



**Rui Jorge Miranda
Araújo**

**Conceptualização e Desenvolvimento de um
Sistema de Gestão de Reclamações**



**Rui Jorge Miranda
Araújo**

Conceptualização e Desenvolvimento de um Sistema de Gestão de Reclamações

Trabalho de Projecto apresentado à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia e Gestão Industrial, realizado sob a orientação científica da Doutora Maria João Machado Pires da Rosa, Professora Auxiliar no Departamento de Economia, Gestão e Engenharia Industrial da Universidade de Aveiro e co-orientação científica da Doutora Leonor da Conceição Teixeira, Professora Auxiliar no Departamento de Economia, Gestão e Engenharia Industrial da Universidade de Aveiro.

o júri

presidente

Prof. Doutor Carlos Manuel dos Santos Ferreira
Professor Associado com Agregação da Universidade de Aveiro

Prof. Doutor José Manuel Matos Moreira
Professor Auxiliar Convidado da Universidade de Aveiro (Arguente)

Prof^a. Doutora Maria João Machado Pires da Rosa
Professora Auxiliar da Universidade de Aveiro (Orientadora)

Prof^a. Doutora Leonor Conceição Teixeira
Professora Auxiliar da Universidade de Aveiro (Co-Orientadora)

agradecimentos

Em primeiro lugar, agradeço aos meus pais que me ofereceram todas as condições para concluir o curso e à minha namorada que esteve sempre presente.

Agradeço de uma forma mais particular à Professora Doutora Maria João, minha orientadora, e Professora Doutora Leonor Teixeira, co-orientadora, que me prestaram apoio incondicional durante a realização do projecto e do estágio curricular.

Agradeço à minha orientadora na empresa de estágio, Eng^a. Cláudia Paiva, que possibilitou a existência deste projecto e, também, ao Eng^o. Nuno Lourenço e Eng^o. Nuno Dias pelo acompanhamento dado na empresa.

Agradeço também aos restantes colaboradores do Departamento de Qualidade de Compras que contribuíram para a minha inserção na equipa e conhecimento dos processos e realidades da empresa, e à empresa *Bosch Security Systems SA* que proporcionou um acompanhamento do seu processo de reclamações.

A todos os demais que contribuíram de alguma maneira para a realização deste projecto.

palavras-chave

Gestão da Qualidade nas Compras, Sistemas de Informação, Benchmarking, Modelação em UML, Bases de Dados, SQL.

resumo

O presente projecto de investigação pretende enfatizar a importância que os Sistemas de Informação e a abordagem por processos têm, numa organização actual, concretamente no apoio à Gestão da Qualidade das Compras, sendo apresentado o desenvolvimento de um Sistema de Informação neste âmbito.

É feita uma pesquisa bibliográfica sobre Gestão pela Qualidade Total e o papel do Benchmarking nesse sentido. São apresentadas as ferramentas da qualidade 8D e A3, como metodologias de resolução de problemas. É abordada a temática do desenvolvimento de Sistemas de Informação e a sua importância, abordando-se também o seu papel como forma de padronização de processos e melhoria contínua, dando-se ênfase às Bases de Dados.

A concretização prática do projecto incide no desenvolvimento, de raiz, de um Sistema de Informação de apoio à Gestão de Reclamações, na organização Bosch Termotecnologia S.A, utilizando para isso UML, e o modelo relacional que é transposto para um Sistema de Gestão de Bases de Dados.

keywords

Purchasing Quality Management, Information Systems, Benchmarking, UML modeling, Databases, SQL.

abstract

The present research project aims to empathize the relevance of the Information Systems and a process approach in organizations, specifically in Purchasing Quality Management support, being presented an information system developed in this context.

A bibliographic research about total quality management is made, including a discussion of the benchmarking concept and methodologies. The 8D and A3 quality management tools are presented as problem solving methodologies. The relevance of Information Systems is discussed, as well as its use as a way to standardize processes and promote continuous improvement, with a special focus in databases development.

The practical component of the Project focus in the development of a Information System to support Complaint Management in the organization *Bosch Termotechnology SA*, using for that UML and a relational model that is translated into a Database Management System.

Índice

1.	Introdução	1
1.1.	Apresentação da Empresa	1
1.2.	Enquadramento do Problema	3
1.2.1.	Gestão da Qualidade das Compras na <i>Bosch Termotecnologia S.A</i>	4
1.2.2.	Processo de Reclamação aos Fornecedores	5
1.2.3.	Ferramenta 8D aplicada às reclamações dos fornecedores	6
1.3.	Objectivos	8
1.4.	Metodologia	8
1.5.	Estrutura do Trabalho	9
2.	Gestão da Qualidade e as Tecnologias e Sistemas de Informação	10
2.1.	Gestão da Qualidade	10
2.1.1.	Gestão Pela Qualidade Total	12
2.1.2.	Sistemas de Gestão da Qualidade	15
2.1.3.	A Norma ISO 9001:2008	16
2.1.4.	Gestão da Qualidade nas Compras	18
2.1.5.	<i>Benchmarking</i>	20
2.1.6.	Ferramentas da Qualidade	27
2.2.	Tecnologias e Sistemas de Informação	31
2.2.1.	Processo de Desenvolvimento de Sistemas de Informação	34
2.2.2.	Bases de Dados e Sistemas de Gestão de Bases de Dados	39
2.2.3.	Projecto de Bases de Dados	40
2.2.4.	Modelos de Bases de Dados	42
2.2.5.	Modelização em UML	46
2.2.6.	Desenho de Interface do Utilizador	49
3.	Desenvolvimento de um Sistema de Gestão de Reclamações: <i>Bosch Termotecnologia S.A</i>	53
3.1.	Análise do Processo Existente	53
3.2.	Análise de Requisitos	55
3.2.1.	Ferramenta A3	55
3.2.2.	<i>Benchmarking</i> e Estudo de Sistemas Existentes	59
3.3.	Solução Proposta	64
3.3.1.	Modelo Conceptual	66
3.3.1.1.	Diagrama de Casos de utilização	67
3.3.1.2.	Diagrama de Classes	69
3.3.1.3.	Diagrama de Sequência	70
3.3.2.	Modelo Lógico	71
3.3.2.1.	Modelo Relacional	72
3.3.3.	Modelo Físico	74
3.3.3.1.	O Protótipo	74
3.3.4.	SQL	80
4.	Conclusões, Limitações e Perspectivas de Trabalho Futuro	82
5.	Referências Bibliográficas	85
6.	Anexos	93
	Anexo 1.Exemplo de Template 8D	94
	Anexo 2.Classificação e Descrição dos diferentes tipos de Diagramas UML	95
	Anexo 3.Ferramenta A3 do Projecto	105
	Anexo 4.Dicionário de dados – diagrama de Classes	106
	Anexo 5.Diagrama de Classes – em versão A3	109
	Anexo 6.Modelo Relacional – em versão A3	110
	Anexo 7.Visual Basic	111

Anexo 8.Exemplo de relatório gerado através de consulta.....	113
Anexo 9.Exemplo de processo do Manual	114
Anexo 10.SQL (<i>Structured Query Language</i>)	115

Índice de Figuras

Figura 1. Enquadramento da <i>Bosch Termotecnologia S.A</i> no Grupo Bosch.....	1
Figura 2. Produtos <i>Bosch Termotecnologia S.A</i>	2
Figura 3. Evolução da Produção.....	2
Figura 4. Market Share <i>Bosch Termotecnologia S.A</i>	2
Figura 5. Dados <i>Bosch Termotecnologia S.A</i>	3
Figura 6. Marcas <i>Bosch Termotecnologia S.A</i>	3
Figura 7. Passos da Ferramenta 8D – Descrição, Ordem e Calendarização.....	7
Figura 8. Planeamento da Qualidade.	11
Figura 10. Infra-estrutura da Gestão pela Qualidade Total.....	15
Figura 11. Ciclo de Deming.....	18
Figura 9. PDCA aplicado a A3.	29
Figura 12. Actividades num Sistema de Informação.	32
Figura 13. Tipos de SI's.	33
Figura 14. Ciclo de Vida dos SI's.....	34
Figura 15. Modelo em Espiral.	35
Figura 16. Modelo em Cascata.	36
Figura 17. Processo Unificado de desenvolvimento.	38
Figura 18. Modelo Relacional.	43
Figura 19. Exemplo Ligação em Modelo Relacional.	43
Figura 20. Exemplo Relação de Um-para-Um.....	44
Figura 21. Exemplo Relação de Um-para-Um.....	45
Figura 22. Exemplo Relação de Muitos-para-Muitos.	45
Figura 23. Evolução da UML.....	47
Figura 24. Hierarquia dos Diagramas UML.	48
Figura 25. Três pilares para a criação de bons interfaces.	50
Figura 26. Diagrama de Actividades – Antes do sistema.....	55
Figura 27. Estado Actual – sem sistema.	56
Figura 28. Estado desejado – futuro.....	57
Figura 29. Acções para implementação.	58
Figura 30. Indicadores de Performance.....	58
Figura 31. Informatização das Reclamações – <i>Bosch Security Systems S.A</i>	60
Figura 32. Diagrama de Actividades – <i>Bosch Security Systems S.A</i>	61
Figura 33. Cadeia do SupplyOn.....	62
Figura 34. Utilização do SupplyOn nas diferentes áreas funcionais.....	62
Figura 35. Diagrama de Actividades - Após Sistema.	66
Figura 36. Diagrama de Casos de utilização para o sistema.....	68
Figura 37. Exemplo de relação <i>extend</i>	68
Figura 38. Exemplo de relação <i>include</i>	69
Figura 39. Diagrama de Classes do Modelo.	70
Figura 40. Diagrama de sequência – Actor Origem a efectuar Reclamação.....	71
Figura 41. Exemplo de transposição – “um para um”.	73
Figura 42. Exemplo de transposição – “um para muitos”.....	73
Figura 43. Modelo relacional – Sistema de Gestão de Reclamações.	74
Figura 44. Menu de Inserção da Reclamação – Origem.....	76
Figura 45. Diagrama de Actividades.....	95
Figura 46. Diagrama de Estado.....	96
Figura 47. Diagrama de Casos de utilização.....	97
Figura 48. Diagrama de Comunicação.....	98
Figura 49. Diagrama de Interacção.....	98

Figura 50. Diagrama de Sequência.....	99
Figura 51. Diagrama de Tempo.....	100
Figura 52. Diagrama de Classes.....	100
Figura 53. Diagrama de Estruturas Compostas.....	101
Figura 54. Diagramas de Componentes.....	102
Figura 55. Diagrama de Objectos.....	102
Figura 56. Diagrama de Pacotes	103
Figura 57. Diagrama de Instalação.....	104
Figura 58. Exemplo de código Visual Basic.....	112

Índice de Tabelas

Tabela 1. Objectivos da Organização – Sem <i>Benchmarking</i> Vs. Com <i>Benchmarking</i>	23
Tabela 2. Tipos de Sistemas de Informação	33
Tabela 3. Actividades no Processo Unificado de Desenvolvimento	39
Tabela 4. Notações Comuns para Relações.....	45
Tabela 5. Balanço <i>Benchmarking</i> - <i>Bosch Security Systems S.A.</i>	60
Tabela 6. SupplyOn Vs. IQUIS – Vantagens e Desvantagens.....	64
Tabela 7. Lista de Casos de utilização.....	67
Tabela 8. Lista de tabelas e chaves primárias.....	72
Tabela 9. Exemplos de print screens da Interface.....	76
Tabela 10. Exemplo de programação associada a botão.....	79

Lista de Abreviaturas

BD –	Base de dados.
DDL –	Data Definition Language.
DML –	Data Manipulation Language.
EDI –	Ethernet Data Interchange.
ER –	Entidade Relação.
ERP –	Enterprise Resource Planning.
FK –	Foreign Key.
GUI –	Graphical User Interface.
IQUIS –	Integrated Quality Improvement System.
ISO –	International Standards Organization.
OMT –	Object Modeling Language.
PPM –	Pieces per Million.
PUQ –	Purchasing Quality.
RUP –	Rational Unified Process.
SGBD –	Sistema de Gestão de Bases de Dados.
SI –	Sistema de Informação.
SQL –	Structured Query Language.
UML –	Unified Modeling Language.

1. Introdução

O presente relatório de projecto inicia-se com uma breve apresentação da organização de estágio. Seguidamente é efectuada uma abordagem ao problema em causa e uma descrição dos processos inerentes ao mesmo. É, também, descrita a metodologia utilizada, sendo este primeiro capítulo concluído com uma breve descrição da estrutura do relatório.

1.1. Apresentação da Empresa

O projecto de investigação descrito neste relatório resulta de um estágio curricular efectuado durante 9 meses na organização *Bosch Termotecnologia S.A*, pertencente ao Grupo Bosch.

O grupo Bosch é detido em 92 % pela Fundação Robert Bosch, tendo sido esta fundada em 1886.

O Grupo iniciou a sua actividade como oficina de electrónica, sendo hoje em dia uma multinacional de referência.

Actualmente o grupo Bosch divide-se em 3 áreas de negócio:

- ✓ Tecnologia Automóvel.
- ✓ Tecnologia Industrial.
- ✓ Bens de Consumo e Equipamentos de Construção.

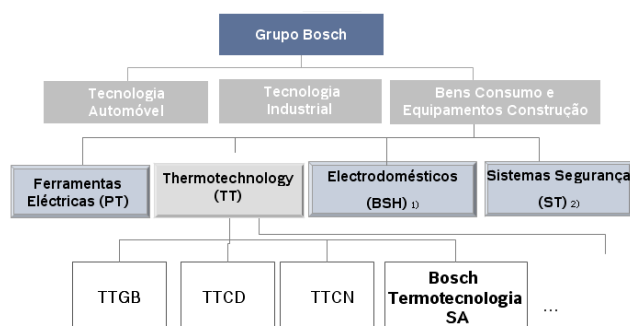


Figura 1. Enquadramento da *Bosch Termotecnologia S.A* no Grupo Bosch (Bosch (2008)).

A *Bosch Termotecnologia S.A*, como o nome indica, pertence à divisão de Termotecnologia (ver figura 1), que por sua vez, pertence ao Grupo de Bens de Consumo e Equipamentos de Construção que representa 26% do volume total de vendas de todo o Grupo (dados relativos a 2007).

A *Bosch Termotecnologia S.A* foi fundada há 32 anos. A sua unidade fabril localiza-se em Aveiro - Cacia, e a direcção comercial em Lisboa.

A oferta de produtos da organização compreende essencialmente 3 áreas (ver figura 2):

- Esquentadores e Termoacumuladores.
- Caldeiras e Aquecimento.
- Solar.



Figura 2. Produtos *Bosch Termotecnologia S.A* (Bosch (2008)).

O produto nuclear são os esquentadores, facto que é facilmente visível quando se analisa a produção nos últimos anos (ver figura 3).

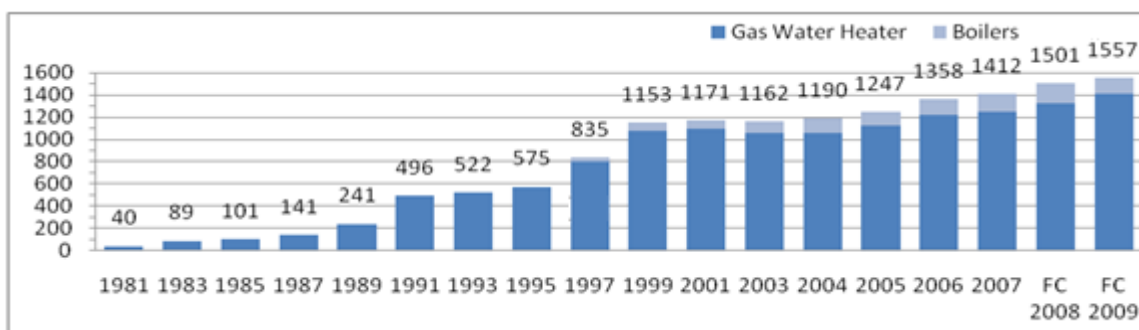


Figura 3. Evolução da Produção (Bosch (2008)).

A *Bosch Termotecnologia S.A* ocupa a posição de líder Europeu de mercado, sendo também a nível mundial uma referência no seu sector, como se pode verificar pelo *market share* nas diferentes regiões (ver figura 4).

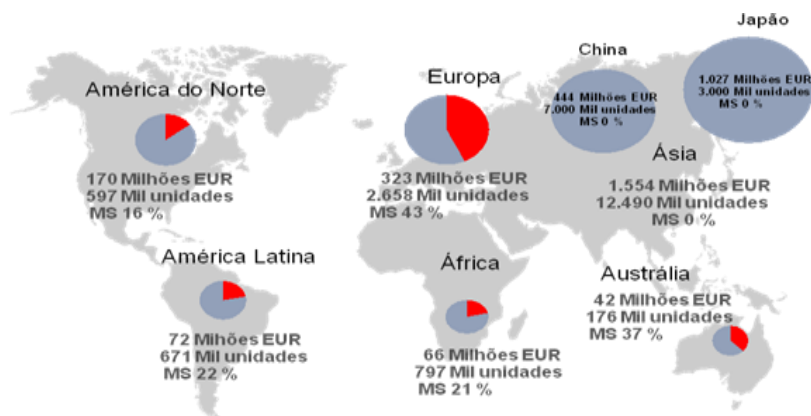


Figura 4. Market Share *Bosch Termotecnologia S.A* (Bosch (2008)).

A análise dos dados (ver figura 5) permite verificar que a organização contava em 2007 com uma capacidade instalada de 1.203.000 esquentadores e 166.000 caldeiras.

O número de colaboradores era de 1133 em 2007. O volume investido em investigação e desenvolvimento tem vindo a aumentar e em 2007 atingiu 5,5 milhões de euros, o que representa 2,3% face à receita de vendas em 2007.

Principais dados	2005	2006	2007
Receita de Vendas (Milhões EUR)	207	230	235
Vendas fora de Portugal (%)			
GWT	77	80	81
GZT	90	92	92
Capacidade de Produção (Milhares)			
GWT	1021	1049	1203
GZT	121	135	166
Investimento (Milhões EUR)	6.6	7.6	7.6
Investigação e Desenvolvimento (Milhões EUR)	3.6	4.5	5.5
Número de Colaboradores	1001	1078	1133

Figura 5. Dados *Bosch Termotecnologia S.A* [2005-2007] (Bosch (2008)).

A inovação e exclusividade dos produtos têm sido uma bandeira da empresa, apresentando a mesma aparelhos com funcionalidades classificadas como USP (*Unique Selling Positions*) graças ao desenvolvimento de novos aparelhos e patentes (ver figura 6).



Figura 6. Marcas *Bosch Termotecnologia S.A* (Bosch (2008)).

A aposta em novos produtos e o nível de serviço ao cliente continuam a ser pontos fundamentais para o sucesso da organização.

1.2. Enquadramento do Problema

Longe vão os tempos em que as organizações produziam todos os componentes afectos aos seus produtos. Em muitas organizações a matéria-prima e componentes, que ingressam a produção, são comprados. Actualmente para a maior parte das empresas, seria impossível produzir todos os componentes que constituem os seus produtos, uma vez que isso implicaria um nível de conhecimento elevado e diversificado e, também, uma exigência muito grande ao nível de recursos, que inviabilizaria a existência da maior parte das organizações.

A *Bosch Termotecnologia S.A* compra inúmeros componentes para os seus produtos, o que torna a gestão da qualidade das compras uma tarefa de elevada importância.

Este facto não se deve apenas ao volume monetário envolvido, mas também aos aspectos legais e de segurança que estão envolvidos.

Ao nível de esquentadores e caldeiras, a responsabilidade civil que a organização tem obriga a que se cumpram alguns requisitos legais, uma vez que, associados ao funcionamento destes aparelhos, estão substâncias inflamáveis e fatais, como o gás e o monóxido de carbono.

Rodriguez & Hemsworth (2005) afirmam que os materiais são uma fonte de variação dos processos e que, por isso mesmo, têm um grande impacto na qualidade dos produtos.

No caso da *Bosch Termotecnologia S.A*, inúmeras peças compradas influenciam directamente o bom funcionamento e a fiabilidade do produto final; como tal, a qualidade destes componentes é de extrema importância para a empresa, uma vez que também o bem-estar do consumidor está em causa com o uso destes produtos.

Rodriguez & Hemsworth (2005) afirmam que, de uma forma geral, as decisões de compra têm grande impacto no produto final das organizações e, consequentemente, no seu desempenho. A *Bosch Termotecnologia S.A*, não é indiferente a estes factores e, rege-se pelos princípios subjacentes à gestão pela qualidade total, apostando numa melhoria contínua de todos os seus processos, produtos, colaboradores e fornecedores.

Reflexo disto é a adopção de diferentes metodologias e técnicas de gestão da qualidade, nomeadamente os *5S*, *6Sigma*, *Lean Production* e *Supplier Development*.

A orientação para a qualidade total, aliada ao seu sistema de produção, que tende cada vez mais para um modelo de *Lean Manufacturing*, implica um estreitamento das relações com os fornecedores, de forma a obter-se o material na hora certa, ao preço mais baixo e com a qualidade desejada.

1.2.1. Gestão da Qualidade das Compras na *Bosch Termotecnologia S.A*

Rodriguez & Hemsworth (2005) afirmam que cada vez mais os gestores vêm as compras e a gestão da qualidade como estando intimamente ligadas.

A existência de *know-how* para gerir a qualidade das compras está presente na *Bosch Termotecnologia S.A*, tendo sido essa a razão pela qual se constituiu o PUQ (*Purchasing Quality*), departamento onde decorreu o presente projecto de investigação.

Inicialmente o PUQ era afecto ao Departamento de Qualidade, mas no decorrer de 2007, de forma a existirem mais sinergias com o Departamento de Compras e a definição de um âmbito comum, foi realizada a sua transferência para este sector.

Das principais funções afectas ao PUQ destacam-se:

- ✓ Avaliação/ Diagnóstico de problemas em peças compradas;
- ✓ Apoio e pedidos de correcção de peças mal desenvolvidas;
- ✓ Débito dos custos de re - trabalho – paragens de linha;
- ✓ Acompanhamento e avaliação de fornecedores – auditorias aos processos dos fornecedores;
- ✓ Desenvolvimento de fornecedores, com o objectivo de prevenir problemas de qualidade – apoio na melhoria dos processos e projectos de melhoria conjuntos;
- ✓ Verificação e bloqueio do *stock* em armazém – garantia da qualidade do mesmo;
- ✓ Elo de ligação *Bosch TT* – Fornecedor;
- ✓ Aprovação de peças novas;
- ✓ Identificação das causas de defeitos nas peças (análise/medição/teste);
- ✓ Apoio à produção em casos de problemas em peças de compra;
- ✓ Re-trabalhos – escolhas após confirmação da causa;
- ✓ Qualidade preventiva – uso de ferramentas da qualidade.

Uma das principais funções do PUQ, que é também uma das mais rotinadas, é o processo de reclamação aos fornecedores e a solicitação da tomada de acções por parte dos mesmos, a qual é descrita com mais detalhe no subcapítulo seguinte.

1.2.2. Processo de Reclamação aos Fornecedores

Todos os dias são efectuadas dezenas de reclamações aos diferentes fornecedores da *Bosch Termotecnologia S.A*, sendo exigido *feedback* das suas acções face aos problemas encontrados.

Como resultado de uma análise ABC¹ às reclamações, que identificou os fornecedores mais problemáticos, a *Bosch Termotecnologia S.A* efectua um acompanhamento rígido aos fornecedores do tipo A e B.

¹ Análise ABC – Método que permite classificar os problemas, pelo seu grau de importância, para consequente tomada de acção, sendo a classificação efectuada em 3 grupos que são o A, B e C. Parte do Pressuposto de Vilfredo Pareto, que dizia que 80% da riqueza estava concentrada em 20% da população. Em termos organizacionais tem a grande valência de permitir detectar rapidamente a pequena parte (exemplo: problemas de qualidade) responsável pela grande parte (exemplo: custos de não qualidade).

No entanto os fornecedores do tipo C não recebem um acompanhamento com o mesmo grau de eficiência, o que acontece dado representarem uma menor quantidade de problemas e custos, isto, apesar de constituírem o grupo maioritário de fornecedores.

Desta forma, a rotina diária faz com que muitas vezes a informação relativa à gestão das reclamações, do tipo C, acabe por se perder em *emails*, fragmentada por múltiplos ficheiros, ou, então, que não chegue a ser exigida ao fornecedor atempadamente.

Por outro lado, existe uma descentralização da informação no que se refere à organização e gestão dos dados e informação afecta às reclamações de um modo geral.

Esta situação origina uma gestão ineficiente dos fluxos de informação inerentes ao processo de reclamação, fomenta a ausência de um histórico e menor controlo das acções pedidas ao fornecedor, o que no limite pode comprometer a qualidade do material comprado.

Num processo de reclamação são normalmente pedidas acções ao fornecedor por meio de uma ferramenta de qualidade, o 8D, abordada no subcapítulo seguinte.

1.2.3. Ferramenta 8D aplicada às reclamações dos fornecedores

Um 8D é uma ferramenta da qualidade constituída por 8 passos (ver ponto 2.1.6.1) que o fornecedor deve seguir, tendo um prazo específico para dar *feedback* dos mesmos à Bosch.

Neste processo a Bosch envia um *template* ao fornecedor de um relatório 8D, parcialmente preenchido, tendo o fornecedor que devolver o 8D totalmente preenchido, num prazo máximo de 20 dias. Esta ferramenta é usada internamente e externamente, ou seja, é utilizada tanto para problemas com fornecedores, como para problemas internos de processo e / ou outros.

O processo 8D associado à reclamação dos fornecedores funciona da seguinte forma: um dia após envio do relatório o fornecedor deve ter a equipa de resolução constituída (D1) e, também, a descrição do problema (D2) efectuada. Ao fim de 2 dias deve ter enviado a definição das acções de contenção (D3), de forma a garantir que efectua uma acção temporária que contenha o problema. Os restantes passos não têm dias específicos, tendo apenas o resultado final do processo que estar na posse da Bosch 20 dias após o requerimento do 8D. Quando o D8 (Reunião Final da Equipa – fecho do 8D), é entregue, o 8D é fechado.

No entanto, de forma a garantir que as acções do fornecedor foram eficazes, é efectuado mais um passo ao fim de 60 dias, que consiste na verificação da eficácia da acção correctiva, tendo sido esta uma melhoria resultante do desenvolvimento do presente projecto de investigação.

Na figura 7 apresentam-se as 8 etapas da ferramenta 8D contemplando os passos e tempos de *feedback* que os fornecedores devem seguir.

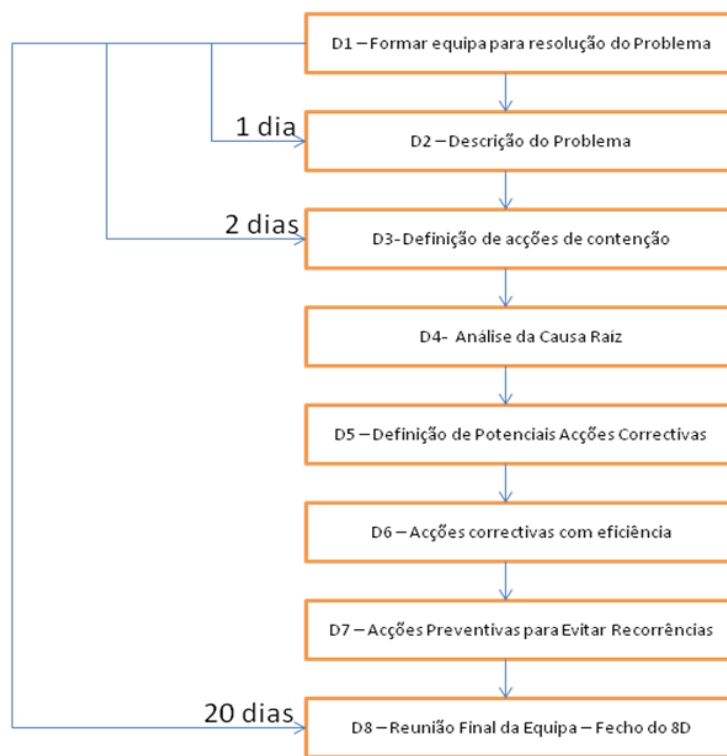


Figura 7. Passos da Ferramenta 8D – Descrição, Ordem e Calendarização.

Apesar de ser um processo simples, muito dos fornecedores atrasam o *feedback* e entrega do relatório, sendo difícil à Bosch monitorizar e controlar os 8D's associados às reclamações, não existindo nenhuma ferramenta para esse efeito que não sejam ficheiros *excel*.

Apesar de a organização efectuar largos investimentos em SI's (Sistemas de Informação) e de estar actualmente numa fase de transição de todos os seus sistemas para um ERP (*Enterprise Resource Planning*)², a situação da Gestão de Reclamações aos Fornecedores é um dos processos que ainda não está contemplado nessa transição. Para além disso, não existem previsões de que essa transição venha a acontecer, uma vez que as soluções padronizadas existentes não têm em consideração as pequenas nuances que determinam e caracterizam o processo de Reclamações da Bosch. Neste sentido surge a necessidade da existência de um Sistema de Informação vocacionado para estes problemas, constituindo esta a principal motivação que determinou o desenvolvimento do presente projecto.

² Sistema de Gestão integrado – sistema que permite estruturar processos e operações de forma integrada / centralizada.

1.3. Objectivos

No sentido de agilizar os mecanismos associados ao processo de reclamações da Bosch, o presente projecto traduz-se, na sua componente prática na análise, conceptualização e desenvolvimento de um sistema de apoio aos fluxos inerentes às reclamações, bem como gestão dos 8D's associados. Em termos de tipologia pode-se caracterizar o sistema como uma Base de Dados que armazena todos o histórico associado às reclamações Bosch, possibilitando, através das suas interfaces gráficas, extracção seleccionada de informação com base em relatórios previamente parametrizados.

Pretende-se implementar este sistema com uma abordagem processual, imprimindo alguns *standards* no processo de reclamações actualmente inexistente.

Com a implementação deste projecto, esperam-se resultados nomeadamente ao nível de diminuição de tempo na Gestão de Reclamações que, na prática, se traduzem numa diminuição de custos associados a este processo e, simultaneamente, na libertação de recursos de modo a que sejam aproveitados para outras tarefas de maior valor acrescentado.

1.4. Metodologia

A metodologia adoptada pode subdividir-se em duas partes distintas. Por um lado tem-se uma componente mais teórica onde se faz uma pesquisa bibliográfica sobre alguns aspectos associados ao tema, de forma a enquadrar teoricamente os principais conceitos que sustentam o projecto. Deste fizeram parte temas como: Gestão da Qualidade, Ferramentas da Qualidade 8D e A3, Gestão da Qualidade dos Fornecedores, Gestão pela Qualidade Total, *Benchmarking*, SI's e sua relação com a Gestão da Qualidade, Modelos de BD's, Linguagem UML, entre outros.

Por outro lado, e após a pesquisa bibliográfica, a metodologia assentou numa vertente mais prática, mais especificamente nas técnicas que suportaram a compreensão do problema, análise do processo de negócios, conceptualização da solução e respectiva implementação. Nesta fase, não só se tentou conhecer o modelo de negócio que suportava o processo de reclamações na Bosch, como também se fez um levantamento de requisitos e necessidades dos potenciais utilizadores do sistema, utilizando-se algumas técnicas, mais especificamente a ferramenta A3, reuniões com os colaboradores e a técnica de *benchmarking*, em que se visitou uma organização e se estudaram alguns sistemas afins existentes no mercado.

Ao longo da análise e modelização fez-se uso da linguagem UML, concretizando-se os resultados em vários modelos (que se traduziram nos diagramas de casos de utilização, actividades, sequência e de classes) vindo estes a definir a estrutura do domínio, e as possíveis interacções com elementos externos do sistema. Por último, e a nível conceptual, obteve-se o modelo relacional que resultou da aplicação de um conjunto de regras de conversão sobre o diagrama de classes.

Na fase de codificação implementou-se o modelo relacional previamente encontrado, utilizando-se para tal um Sistema de Gestão de Base de Dados (SGBD) em *Microsoft Access* (versão de 2003), com apoio de uma ferramenta de programação *Microsoft Visual Basic* (para se efectuarem as consultas necessárias e impor as condições restritivas e/ou obrigatórias, respectivamente). A interface foi construída de forma centrada no utilizador, tendo este um papel importante na definição do resultado final.

Posteriormente preparou-se uma formação para os colaboradores sobre a utilização e manutenção do sistema, sendo também o sistema testado em ambiente real pelos mesmos utilizadores.

Por fim, foi construído o manual de utilização do Sistema com vista a detalhar ao máximo todos os processos de utilização e a potenciar de forma o mais eficiente possível a sua utilização.

1.5. Estrutura do Trabalho

O presente relatório de projecto encontra-se estruturado em quatro capítulos. No primeiro é feita uma introdução, com uma breve descrição da empresa onde decorreu todo este projecto, a contextualização do problema em estudo, principais objectivos e descrição da metodologia adoptada no sentido de se alcançarem tais objectivos.

No segundo capítulo é feito o enquadramento teórico subjacente ao projecto desenvolvido, dando-se ênfase aos SI's e à sua ligação às questões da Gestão pela Qualidade Total e *Benchmarking*.

No terceiro capítulo temos a descrição da solução, concretizada através de um Sistema de Gestão de Reclamações, bem como todo o seu processo de desenvolvimento.

Por último, no capítulo das conclusões, são retiradas as principais ilações do trabalho, e algumas limitações que reverteram para práticas e propostas de trabalho futuro.

2. Gestão da Qualidade e as Tecnologias e Sistemas de Informação

Actualmente a Qualidade e os SI's são temas em destaque nas organizações. Cada vez mais organizações se regem pela melhoria dos seus processos e fluxos, baseando-se em princípios de Gestão pela Qualidade Total que, muitas vezes, implicam a adopção de SI's.

Assim sendo, estas são disciplinas que, embora distintas, podem estar intimamente ligadas.

Apesar de o presente projecto de investigação ser bastante orientado para o desenvolvimento de SI's, a gestão da qualidade está presente em toda a sua natureza uma vez que é desenvolvido no âmbito da melhoria contínua de processos, neste caso do processo de Gestão de Reclamações aos Fornecedores.

Importa, portanto, perceber os conceitos de qualidade e de melhoria contínua, bem como a sua importância ao nível da Gestão de Fornecedores, âmbito no qual o SI é desenvolvido.

2.1. Gestão da Qualidade

Existe um vasto conjunto de definições de qualidade, não havendo uma concordância e um significado universal para o conceito. No entanto, muito dos autores referem-se à qualidade como sendo a capacidade de uma organização conhecer e responder aos requisitos dos seus clientes.

De acordo com Karahasanovic & Lonn (2007), a qualidade representa a capacidade de preencher as expectativas de um mercado ou cliente, sendo a mesma traduzida em serviços ou produtos que satisfazem o cliente.

Oakland & Sohal (2001) referem-se à qualidade como sendo, simplesmente, o conhecimento dos requisitos do cliente.

Verifica-se, portanto, que subjacente ao conceito de Qualidade está a orientação para o cliente, o saber o que o cliente quer. Muitas vezes este é um exercício difícil para uma organização, tanto ao nível de cliente externo como interno.

Oakland & Sohal (2001) afirmam que de forma a alcançar a qualidade ao longo de toda a organização, cada pessoa tem que se interrogar sobre o seguinte:

- ✓ Quem são os meus clientes alvo?
- ✓ Quais são os seus requisitos?
- ✓ Como posso saber quais são os seus requisitos?

- ✓ Como posso medir a minha capacidade de medir os requisitos?
- ✓ Tenho as capacidades necessárias para conhecer os requisitos? Se não, o que posso fazer para as alcançar?
- ✓ Conheço os requisitos de uma forma contínua?
- ✓ Como monitorizo as alterações dos requisitos?

No entanto, a qualidade é mais do que isto. Para se alcançar a qualidade tem que existir um processo de gestão da mesma.

Juran & Godfrey (1998), dividem a gestão da qualidade em 3 processos distintos:

- ✓ **Planeamento da Qualidade** – Começa-se por identificar os clientes e as suas necessidades: seguidamente concebem-se os produtos (bens e/ou serviços) que respondam às necessidades levantadas. Posteriormente concebem-se os processos que podem produzir os produtos e finalmente, coloca-se o plano em acção e controlam-se as operações (ver figura 8).



Figura 8. Planeamento da Qualidade (Adaptado de Juran & Godfrey (1998)).

Os autores enfatizam que por muito que se apliquem ferramentas da qualidade e métodos de controlo, a maioria dos processos admite variabilidade, existindo problemas crónicos de desperdício, tais como tempo, atrasos, re-trabalhos, etc. Têm, então, que se providenciar sistemas de controlo, de forma a manter os níveis de qualidade como planeado. Seguidamente têm que se procurar oportunidades de melhoria.

- ✓ **Controlo da Qualidade** – Operações para redução do desperdício. O controlo da qualidade deve assentar numa clara definição de qualidade, focalização num resultado específico, um objectivo claro, uma forma de medir e interpretar o desempenho actual e comparar com o resultado pretendido, e uma forma de tomada de acção e de ajuste do processo se necessário.

- ✓ **Melhoria da Qualidade** – Toda a actividade anterior mantém a qualidade no nível planeado. Têm que ser tomadas acções específicas para mudar esse nível (para um superior), o que significa melhorar a qualidade.

2.1.1. Gestão Pela Qualidade Total

A gestão pela qualidade total é uma área de gestão muito abordada e que se encontra presente em algumas organizações como por exemplo a Xerox, Procter & Gamble, Motorola, Ford Motor Corp., Saturn Corporation, Florida Power and Light (Anjard, 1998).

As organizações preocupam-se cada vez mais não só com a qualidade dos bens, produtos e serviços tendo em consideração os seus clientes, mas também com todas as suas actividades e processos, desde a entrada à saída, e com a qualidade dos seus parceiros, quer como fornecedores, quer como distribuidores.

Rodriguez & Lorente (2004) afirmam que o conceito de gestão pela qualidade total surgiu por volta de 1980, como resposta a uma procura de baixos custos, qualidade elevada e produtos fiáveis, com design melhor e maior flexibilidade e que este conceito veio modificar a forma como as organizações gerem a qualidade.

Os mesmos autores dizem que este conceito levou a que as organizações não se focassem exclusivamente em actividades de detecção, para assegurar a qualidade dos produtos, mas sim na implementação de novas actividades com o objectivo de reduzir e eliminar causas de qualidade defeituosa.

Neste sentido a Gestão pela Qualidade Total dá uma outra dinâmica à Gestão da Qualidade, uma vez que é uma tarefa que se baseia na acção e implementação de novas ideias e conhecimentos, envolvendo toda a organização.

Madu et Al. (1994) afirmam que no ambiente competitivo que existe actualmente, a chave para o sucesso organizacional assenta na qualidade, devendo as organizações incorporar os princípios da gestão pela qualidade total em todas as suas actividades, dando ênfase à filosofia da melhoria continua.

Omachou & Ross (2004) dizem que a gestão pela qualidade total consiste na integração de todas as funções e processos numa organização, de modo a alcançar uma melhoria contínua da qualidade dos bens e serviços prestados, sendo o objectivo último a satisfação do cliente. O autor foca a necessidade de as organizações se centrarem na melhoria contínua dos seus processos sendo esse o princípio principal subjacente à gestão pela qualidade total.

Anjard (1998) afirma que para combater a obsolescência organizacional as empresas devem implementar um sistema de gestão pela qualidade total. Diz também que através da gestão pela qualidade total se encorajam os colaboradores a ser responsáveis pela entrega de qualidade nos seus produtos e serviços, sendo também alcançados os seguintes resultados aquando da sua implementação com sucesso:

- ✓ Envolvimento de todos na melhoria continua.
- ✓ Menos discussões e re-trabalhos, maior esforço para eliminar causas raiz de problemas e melhor histórico.
- ✓ Maior esforço para clarificar os requisitos e prevenir proactivamente a recorrência de defeitos e erros.
- ✓ Maior abertura e comunicação entre pessoas que lidam com os clientes e fornecedores.
- ✓ Maior focalização nos processos de trabalho e sua melhoria.
- ✓ Problemas identificados e resolvidos a níveis menores – *empowerment*.
- ✓ Intolerância perante erros que anteriormente se achavam comuns e geravam indiferença.

Rodriguez & Lorente (2004) afirmam que se uma organização quiser alcançar a qualidade total necessita de mobilizar todos os seus recursos, internos e externos, num processo contínuo de melhoria de produtos e serviços; os autores referem que se as organizações tiverem um comprometimento verdadeiro com a gestão pela qualidade total, a qualidade dos materiais comprados e dos processos internos têm que se tornar tão importantes como a qualidade dos produtos e serviços disponibilizados para os clientes externos.

Estes autores enfatizam ainda a necessidade da organização se melhorar continuamente e da existência de um comprometimento com a qualidade dos materiais comprados e dos processos.

Anjard (1998) afirma que a qualidade total se rege pela concepção dos produtos e processos, de forma a maximizar a satisfação dos clientes ao mínimo custo possível.

Omachonu & Ross (2004) abordam a gestão pela qualidade total como sendo um processo afecto a todos os níveis da organização, sendo necessário o envolvimento de todos para se conseguir a sua implementação.

Anjard (1998) afirma que a implementação da gestão pela qualidade total segue 7 passos, pela seguinte sequência:

- i) Princípios da Gestão – bases, regras e necessidades.
- ii) Visão – desenvolver a declaração da visão.
- iii) Identificar os clientes internos e externos – as suas necessidades.
- iv) Constituir uma equipa.
- v) Treinar a equipa.
- vi) Estabelecer um método de medição.
- vii) Estabelecer um ciclo de revisão e melhoria contínua – desenvolver objectivos tangíveis.

Com a implementação destes 7 passos podem ser obtidos aumentos no retorno e redução de custos.

O autor afirma também que o principal investimento financeiro para adoptar a gestão pela qualidade total é o investimento na formação dos colaboradores. O papel dos colaboradores na gestão pela qualidade total é essencial. São eles que têm que estar predispostos a melhorar os processos e produtos, pois são eles que melhor os conhecem.

Anjard (1998) afirma que a gestão e cultura da organização são os factores mais determinantes no sucesso da implementação de um sistema de gestão pela qualidade total, afirmando que expressões como *empowerment*, trabalho de equipa, gestão colaborativa, envolvimento 100% na qualidade, aprendizagem contínua, conhecimento das ferramentas da qualidade, políticas fortes de recursos humanos e respeito por indivíduos, têm que estar presentes nesta filosofia.

O estudo de Costa & Jimenez (2008) vem apoiar o que os autores já referidos afirmam, tendo concluído, com base numa amostra 1600 organizações, que existe maior benefício para as organizações que aplicam os conceitos da gestão pela qualidade total do que para aqueles que não o fazem.

Juran & Godfrey (1998) apresentam o que consideram ser a infra-estrutura da gestão pela qualidade total (ver figura 10).

Juran & Godfrey (1998) focam alguns dos pontos já abordados até agora; no entanto dão especial ênfase aos sistemas de qualidade e ao desenvolvimento das relações com os clientes e fornecedores.



Figura 9. Infra-estrutura da Gestão pela Qualidade Total (Adaptado de Juran & Godfrey (1998)).

Os autores avançam com o exemplo da família de normas ISO 9000, que dizem fornecer um bom ponto de partida para o estabelecimento de relações de parceria com clientes e fornecedores.

2.1.2. Sistemas de Gestão da Qualidade

Os sistemas de gestão da qualidade têm vindo a assumir um papel cada vez mais importante nas organizações. Actualmente, numa era em que o mercado é global, as organizações apostam nos sistemas de gestão da qualidade não só para adoptarem uma metodologia sistematizada perante a qualidade, mas como forma de passar a imagem ao mundo de que apostam seriamente na qualidade dos seus produtos e serviços. Para que isto aconteça muitas das organizações certificam os seus sistemas de gestão da qualidade, junto de entidades com a competência para tal.

Stahl (2006) afirma que as normas ISO 9000 são as normas mais populares na área de melhoria contínua da qualidade. Para Juran & Godfrey (1998) as normas ISO 9000 podem resumir-se à expressão “globalização do negócio”, e que a sua existência se deve aos seguintes acontecimentos: novas tecnologias, aumento da população mundial, *downsizing* das organizações, *outsourcing*, diversidade cultural e linguística, emergência de novas potências e novos tipos de mercados e concorrências.

As normas ISO 9000 proporcionam um padrão reconhecido em todo o mundo, sendo por isso tão utilizadas actualmente, até porque obedecem a uma linguagem global.

Juran & Godfrey (1998) afirmam que a família de normas ISO 9000 são escritas em termos do que os sistemas da gestão da qualidade devem ter, e não em termos de como se deve fazer para implementar esses requisitos. Ou seja, baseiam-se num conjunto de orientações que as organizações devem seguir. A Apcer (2003) retrata este facto, referindo no seu guia interpretativo da norma ISO 9001:2000 o seguinte:

“As normas da família ISO 9000 são referenciais para a implementação de sistemas de gestão da qualidade (SGQ) que representam um consenso internacional sobre boas práticas de gestão e com o objectivo de garantir, da primeira e de todas as vezes, o fornecimento de produtos que satisfaçam os requisitos dos clientes ou estatutários e/ou regulamentares, bem como a prevenção dos problemas e a ênfase na melhoria contínua.”

Juran & Godfrey (1998) referem um ponto importante quando afirmam que ao implementar a família de normas ISO 9000 as organizações utilizam fluxogramas e outros métodos que enfatizam o trabalho por processos aproveitando oportunidades para simplificação e optimização de processos.

As normas ISO 9000:2000 são constituídas por 3 normas principais:

ISO 9000 - Sistemas de Gestão da Qualidade. Fundamentos e Vocabulário;

ISO 9001 - Sistemas de Gestão da Qualidade. Requisitos;

ISO 9004 - Sistemas de Gestão da Qualidade. Linhas de Orientação para a Melhoria do Desempenho.

2.1.3. A Norma ISO 9001:2008

A norma ISO 9001:2008 é constituída por requisitos padrão a que uma organização deve obedecer para a implementação de um sistema de gestão da qualidade que pode, posteriormente, certificado. Juran & Godfrey (1998) afirmam que a norma fornece um conjunto de requisitos que servem como de linhas de orientação para as organizações.

São inúmeras as organizações que possuem certificação do seu sistema de gestão da qualidade pela norma ISO 9001:2008.

Esta norma baseia-se nos 8 princípios da Gestão da Qualidade, descritos na norma ISO 9000:2000, e que se apresentam seguidamente:

- ✓ **Focalização no Cliente** – Este princípio baseia-se na dependência que as organizações têm dos seus clientes devendo, por isso, compreender as suas necessidades presentes e futuras.
- ✓ **Liderança** – Os líderes estabelecem a finalidade e orientação da organização, devendo manter um ambiente interno que permita o desenvolvimento das pessoas para que se atinjam os objectivos.

- ✓ **Envolvimento das Pessoas** – As pessoas são a essência da organização e, como tal, o seu envolvimento permite que as suas capacidades sejam orientadas para a organização. As pessoas e a sua formação são essenciais para a organização.
- ✓ **Abordagem por processos** – Este princípio baseia-se no pressuposto de que um resultado desejado é atingido de forma mais eficiente quando as actividades e os recursos associados são geridos como um processo. Enfatiza a necessidade de as organizações assumirem as suas actividades como *inputs* que são processados e geram *outputs*.
- ✓ **Abordagem da gestão como um sistema** - Identificar, compreender e gerir os processos inter-relacionados como um sistema, contribui para que a organização atinja os seus objectivos com eficácia e eficiência.
- ✓ **Melhoria contínua** – Convém que a melhoria contínua do desempenho global de uma organização, seja um objectivo permanente. A melhoria contínua deve ser feita a todos os níveis na organização incluindo processos, produtos e pessoas.
- ✓ **Abordagem à tomada de decisões baseada em factos** - As decisões eficazes são baseadas na análise de dados e de informações. A tomada de decisões deve ser feita com base em dados.
- ✓ **Relações mutuamente benéficas com fornecedores** - Uma organização e os seus fornecedores são interdependentes e uma relação de benefício mútuo potencia a aptidão de ambas as partes para criar valor.

A cláusula 4.1 – Requisitos gerais, da norma ISO 9001:2008 refere a pertinência da adopção de uma abordagem por processos, referindo-se, num sub-ponto, à necessidade de existência de informação e recursos que possam suportar a monitorização dos processos. Neste sentido a utilização de Sistemas de Informação poderá ser uma mais-valia para as organizações que procuram implementar sistemas de gestão da qualidade.

De acordo com a norma ISO 9000:2008 espera-se que uma organização aplique uma abordagem PDCA (planear – fazer – verificar – agir) aos seus processos.



Figura 10. Ciclo de Deming (Sinfic (2005)).

De uma forma muito geral, pode dizer-se que os principais contributos que a norma ISO 9001:2008 traz são a abordagem por processos que sugere o ciclo de melhoria contínua de Deming.

2.1.4. Gestão da Qualidade nas Compras

Nos últimos anos a área das Compras tem vindo a ganhar um papel cada vez mais importante e dinâmico. As compras já não são vistas apenas como uma actividade de transacção e comunicação com o fornecedor.

Stanley & Wisner (2000), enfatizam esta mudança, afirmando que o papel das compras se alterou significativamente nos últimos 15 anos, passando de orientação nas transacções para orientação para a gestão da cadeia de abastecimento e salientam que os gestores de compras cada vez mais reconhecem o cliente externo como seu cliente.

Os mesmos autores afirmam que à medida que a competição aumenta os requisitos do produto e as expectativas de qualidade tendem a aumentar também, resultando em especificações de compra e requisitos de fornecimento mais rigorosas, o que leva a um aumento da complexidade da função das compras.

Car et All. (2002) dividem as compras em estratégicas e não estratégicas:

- ✓ **Compra Estratégica** – quando é proactiva e está alinhada com os objectivos da organização, tem âmbito de longo-prazo e cariz integrativo.
- ✓ **Compra Não estratégica** – quando é reactiva para outras funções, não integrativa e se foca em questões de curto-prazo.

Os autores exemplificam as funções de compras não estratégicas como sendo processamento de ordens e expedição de ordens de compra de outros departamentos, tendo isto um peso apenas ao nível operacional.

Referem ainda que este tipo de compras não oferece desenvolvimento técnico e pessoal aos seus colaboradores e que normalmente é pouco informatizada sendo resultante, normalmente, de uma quantidade de compras que envolve pouco capital e, por isso, não lhe é atribuída muita importância.

Por outro lado, atribuem às compras estratégicas uma função dinâmica e de responsabilidade, que é vista pela gestão de topo como um meio de acrescentar valor ao produto e diminuir custos através da gestão de custos. Neste tipo de compra os colaboradores são proactivos e são-lhes incutidas formação e especialização, sendo considerados recursos da organização que a levam a ganhar vantagem competitiva

O autor realça a orientação para o cliente, que tendencialmente tem vindo a existir no papel das compras. É importante saber o que o cliente quer para se comprar o material certo e com a qualidade certa.

Carr et al. (2002) afirmam que actualmente as empresas têm que adquirir estrategicamente os materiais e serviços que lhes permitam satisfazer os requisitos dos seus clientes.

Como já foi referido o conhecimento dos requisitos do cliente é fundamental para a qualidade, e esse conhecimento passa também por toda a cadeia de abastecimento.

Stanley & Wisner (2000) referem que actualmente o papel das compras é importante como intermediário entre os clientes externos e os clientes internos, e que é uma das actividades que ultrapassa os limites internos da organização e serve de fornecedor interno da organização.

As compras seguem o paradigma de orientação para o mercado, uma vez que cada vez mais estão focalizadas no que o cliente quer, isto é importantíssimo uma vez que as organizações satisfazem os seus clientes em grande parte devido a componentes que podem provir dos seus fornecedores.

Para Carr et al. (2002) à medida que as organizações têm vindo a perceber a importância dos *inputs* comprados para os seus produtos, o papel das compras no planeamento estratégico organizacional tem vindo a aumentar também.

Stanley & Wisner (2000) afirmam que as compras têm um papel fulcral na integração dos canais de abastecimento externos e internos, sendo um factor chave determinante na transformação, com sucesso, dos materiais comprados em produtos finais e serviços.

Os mesmos autores concluíram, com base na realização de um trabalho de investigação sobre esta temática, que a implementação de relações colaborativas entre compradores e fornecedores está associada positivamente à capacidade de os departamentos de compras entregarem material de qualidade aos seus clientes internos. Perante os resultados da investigação realizada é pacífico afirmar que as relações com os fornecedores têm um papel cada vez mais importante na melhoria da qualidade dos produtos das empresas.

De acordo com Rodriguez & Hemsworth (2005) as práticas de gestão de qualidade aplicadas às compras, tais como a gestão da qualidade dos fornecedores, a gestão de recursos humanos, a coordenação inter-departamental, a qualidade da informação e o *benchmarking*, contribuem para o melhor desempenho das compras, o que se traduz em, aumento da qualidade dos materiais comprados, maior cumprimento dos *lead-times*, alcance dos objectivos de inventário e cumprimento das expectativas de gastos de material. Estes autores afirmam, também, que a melhoria do desempenho das compras tem influência no desempenho global das organizações. Pode, portanto, afirmar-se, ainda que indirectamente, que a existência de práticas de gestão da qualidade nas compras influencia o desempenho das organizações.

Stanley & Wisner (2000) afirmam que a qualidade ao longo de toda a cadeia de abastecimento pode ter um impacto positivo na satisfação do cliente e ser importante no alcance de *market share* para todos os membros da cadeia.

Para Rodriguez e Lorente (2004) a gestão da qualidade nas compras é constituída por 6 factores principais que são, o suporte da gestão de topo, gestão de recursos humanos, gestão de processos, relações com o cliente, gestão da qualidade dos fornecedores, qualidade da informação e concepção do produto.

Hemsworth et al. (2005) afirmam que as práticas de gestão de qualidade nas compras, como sejam a gestão de recursos humanos, a coordenação interdepartamental, o comprometimento da gestão de topo, o *benchmarking* e o desenvolvimento de SI's aumentam consideravelmente o desempenho da função.

2.1.5. *Benchmarking*

O termo *benchmarking* é actualmente utilizado com alguma frequência. De uma forma geral, pelo senso comum, atribui-se a este termo o significado de estudo das melhores práticas de organizações parceiras.

No entanto, o conceito de *benchmarking* é significativamente mais abrangente e complexo.

O *Benchmarking* significa aprender com os outros e copiar as melhores práticas para facilitar a melhoria contínua na organização (Dale, 1996).

Omachonu & Ross (2004) afirmam que o *benchmarking* é um método pelo qual se podem identificar novas ideias e novas formas de melhorar os processos, indo ao encontro das expectativas dos clientes.

Oakland & Sohal (2001) são mais específicos e afirmam que o *benchmarking* é um processo contínuo de medição dos produtos, serviços e processos comparativamente aos líderes de indústria ou concorrentes mais fortes, resultando numa busca das melhores práticas que levam a um desempenho superior, alcançado através da medição da desempenho e implementação da mudança.

Freytag & Hollensen (2001) abordam o *benchmarking* como uma forma de identificar as melhores práticas que podem ser adoptadas e implementadas pela organização, com o propósito de melhorar o seu desempenho, sendo uma prática desenvolvida, normalmente, dentro de indústrias semelhantes.

Chen (2001) afirma que o *benchmarking* é utilizado por muitos sectores da economia, enquanto ferramenta para controlos de custos e qualidade do produto.

Para Oakland & Sohal (2001) o *benchmarking* tem como objectivo retirar o melhor das pessoas, processos, produtos, fábricas e programas. Deve ser uma metodologia de melhoria contínua, não devendo ser usada isoladamente; uma vez iniciada deve existir sempre uma procura das melhores práticas, de forma a melhorar continuamente (Freytag & Hollensen, 2001).

Na literatura existe o consenso geral de que o *benchmarking* se traduz numa melhoria por aprendizagem, mudança e comparação com algo, quer seja de forma intrínseca, ou extrínseca, à organização.

O *benchmarking* teve origem no Japão e foi utilizado de uma forma pioneira pela Xerox, no âmbito de um programa de liderança pela qualidade nos anos 80 (século XX) (Chen (2001), Freytag & Hollensen (2001), Oakland & Sohal (2001), Omachonu & Ross (2004), Zairi (1994)).

A partir daí tornou-se uma técnica competitiva utilizada por muitas organizações, tais como a IBM, Motorola, AT & T, 3M, DuPont, como forma de minimizar os custos de produção, defeitos e para melhorar a produtividade e conhecer os requisitos dos clientes (Chen, 2001). O CEO da Xerox, David Kearns, definiu a abordagem pelo *benchmarking* como uma forma de melhorar continuamente os produtos e, de forma a competir com os melhores e os líderes de indústrias (Chen, 2001).

Oakland & Sohal (2001) enfatizam a importância do *benchmarking* e do seu âmbito como ferramenta de qualidade e estratégica, afirmando que a sua utilização deve ter como objectivo principal a produção de produtos e serviços consoante os requisitos do cliente, sendo essencial o seu uso no sentido de melhoria contínua com todos os departamentos envolvidos.

Verifica-se, portanto, que o *benchmarking* é uma ferramenta enquadrada nos princípios da gestão pela qualidade total, assentando num pressuposto de melhoria contínua.

Freytag & Hollensen (2001) focam este ponto ao afirmar que o *benchmarking* só faz sentido quando aplicado de forma contínua e assente num princípio de melhoria contínua, sendo parte integrante de uma organização regida pelos princípios de gestão pela qualidade total.

Também para Chen (2001) o *benchmarking* é um componente das técnicas modernas de gestão, sendo parte da gestão pela qualidade total. Zairi (1994), afirma que o *benchmarking* apenas pode ser aplicado como parte integrante da gestão pela qualidade total e, como tal, direccionado para organizações que utilizam ferramentas, sistemas, trabalham em equipas multi-disciplinares e utilizam uma abordagem de resolução de problemas. Este autor refere que o *benchmarking* pode ser efectuado por qualquer organização desde que tenha um comprometimento contínuo em melhorar.

Existe, portanto, o consenso de que o *benchmarking* está associado à gestão pela qualidade total e às organizações que procuram melhorar os seus processos, serviços ou produtos.

Freytag & Hollensen (2001) definem três conceitos distintos para o que a generalidade dos autores consideram ser apenas o *benchmarking*, apresentando além deste, mais dois conceitos distintos, mas relacionados, que são:

- ✓ **Benchlearning:** processo de aprendizagem através dos melhores, com o propósito de integrar essas práticas a todos os níveis da organização.
- ✓ **Benchaction:** implementação actual, das mudanças planeadas, na organização.

Que vantagens é que uma organização pode obter com a implementação do *benchmarking*? Importa saber, especificamente, o que se pode obter com o *benchmarking* e, neste sentido, Oakland & Sohal (2001) apresentam alguns dos principais objectivos pelos quais vale a pena utilizar esta metodologia (ver tabela 1).

Tabela 1. Objectivos da Organização – Sem *Benchmarking* Vs. Com *Benchmarking* (Adaptado de Oakland & Sohal (2001)).

Objectivos	Sem <i>Benchmarking</i>	Com <i>Benchmarking</i>
• Tornar-se competitivo.	<ul style="list-style-type: none"> Focalização interna. Mudança baseada na evolução. 	<ul style="list-style-type: none"> Compreensão da competitividade. Ideias de práticas reais.
• Melhores práticas da indústria.	<ul style="list-style-type: none"> Poucas soluções. 	<ul style="list-style-type: none"> Muitas opções. Desempenho superior.
• Definição dos requisitos do consumidor.	<ul style="list-style-type: none"> Baseado no histórico. Percepção. 	<ul style="list-style-type: none"> Realidade do mercado Avaliação objectiva.
• Estabelecimento de objectivos efectivos.	<ul style="list-style-type: none"> Reactivo. Falta de focalização no exterior. 	<ul style="list-style-type: none"> Credibilidade Proactivo.
• Desenvolvimento de medidas de produtividade.	<ul style="list-style-type: none"> Forças e fraquezas não percepcionadas. Pouco esforço. 	<ul style="list-style-type: none"> Resolução de problemas reais. Compreensão dos outputs Baseada nas melhores práticas da indústria.

Omachonu & Ross (2004), por outro lado, não efectua uma comparação entre objectivos com *benchmarking* e sem *benchmarking*, mas indicam alguns dos principais benefícios de quem o efectua:

✓ **Mudança Cultural:**

Permite às organizações definir novos objectivos, realísticos, de desempenho. Enfatiza o olhar para outras organizações para mostrar que é possível fazer, adoptar e implementar novas soluções.

✓ **Melhoria do Desempenho:**

Permite às organizações identificar falhas no seu desempenho e seleccionar processos para as melhorar. Proporciona um meio de os produtos e serviços serem re-concebidos para alcançar as expectativas dos clientes.

✓ **Recursos Humanos:**

O *Benchmarking* fornece uma base para formação. Os colaboradores começam a verificar a discrepância entre o que eles fazem e o que os melhores fazem. Colmatar essa discrepância requer o envolvimento das pessoas nas técnicas de resolução de problemas e melhoria dos processos. A sinergia entre actividades organizacionais é melhorada através da cooperação interdepartamental.

De tudo o que foi referido pode, portanto, concluir-se que o *benchmarking* pode ser muito útil às organizações; no entanto, é importante saber como o efectuar e com quem.

Neste sentido vários autores (Oakland & Sohal, 2001 e Freytag & Hollensen, 2001) apresentam uma divisão dos tipos de *benchmarking* existentes, apresentando uma diferenciação que pode proporcionar uma abordagem mais metodológica e focalizada:

- ✓ **Interno** – Comparações de operações internas (Oakland & Sohal, 2001), uma vez que a maioria das organizações possuem funções similares dentro da sua unidade de negócio (Freytag & Hollensen, 2001). O benefício imediato reside na identificação dos melhores procedimentos internos e sua disseminação para outras partes da organização (Freytag & Hollensen, 2001). Freytag & Hollensen (2001) afirmam que este tipo de *benchmarking* pode, por vezes, levar as organizações a uma visão demasiado interna, sendo o ideal este tipo de *benchmarking* servir de base para o *benchmarking* externo.
- ✓ **Competitivo** – Comparações competitivas de um produto ou função, com base num concorrente específico (Oakland & Sohal, 2001). Freytag & Hollensen (2001) afirmam que este tipo de *benchmarking* é utilizado contra competidores directos e que o seu objectivo principal é comparar organizações que oferecem produtos, serviços ou processos nos mesmos mercados.
- ✓ **Funcional (Industrial)** – Comparações de funções similares dentro da mesma indústria ou com líderes de indústria. (Oakland & Sohal, 2001). Para Freytag & Hollensen (2001) o *benchmarking* funcional baseia-se na medição de alguma variável organizacional interna, e sua posterior comparação com variáveis similares de outras organizações, normalmente líderes industriais. Estes mesmos autores afirmam que os parceiros de *benchmarking* partilham normalmente alguma tecnologia comum e características de mercado: uma vez que não são concorrentes directos os parceiros de *benchmarking* tem mais vontade de contribuir e partilhar.
- ✓ **Genérica (Processo)** - Comparações de processos de negócio muito similares. (Oakland & Sohal, 2001). Freytag & Hollensen (2001) abordam este tipo de *benchmarking* como sendo difícil de efectuar e implementar.

Dizem que é aplicado a processos semelhantes em organizações que podem ser bastante diferentes e, mesmo, de indústrias distintas, requerendo um conhecimento profundo dos processos a estudar.

Para Freytag & Hollensen (2001) o *benchmarking* é uma técnica de gestão estratégica para obter melhorias de desempenho operacionais e estratégicas. Como qualquer ferramenta estratégica e de gestão tem que existir uma abordagem e metodologia a seguir, as quais se descrevem seguidamente.

2.1.5.1. Como efectuar *Benchmarking*

Diversos autores dão contributos neste sentido (Freytag & Hollensen (2001), Zairi (1994), Chen (2001)). No entanto, Oakland & Sohal (2001) dividem o processo de *benchmarking* em 15 passos simples que representam, de uma forma estruturada, uma abordagem possível.

Planeamento:

- ✓ Selecção do departamento ou processo para *benchmarking*.
- ✓ Identificação do melhor competidor, utilizando o *feedback* dos clientes ou observadores industriais.
- ✓ Identificação de *benchmarks*.
- ✓ Formação de equipa apropriada.
- ✓ Decisão sobre o método de recolha de dados e a metodologia a utilizar.
- ✓ Preparação de visitas e interacção com as organizações alvo.
- ✓ Uso de metodologias de recolha de dados.

Análise

- ✓ Comparação da organização com os seus concorrentes utilizando os dados de *benchmarking*.
- ✓ Catalogação da informação e criação de centro de competências.
- ✓ Compreensão dos processos e das medidas de desempenho.

Desenvolvimento

- ✓ Desenvolvimento de novos objectivos de desempenho.
- ✓ Desenvolvimento de planos de acção para alcançar objectivos e sua integração na organização.

Melhoria

- ✓ Implementação de acções específicas e sua integração nos processos de negócio.

Revisão

- ✓ Monitorização dos resultados e melhorias.
- ✓ Revisão dos *benchmarks* e das relações com as organizações alvo.

O *benchmarking*, tal como outras ferramentas, tem as suas limitações. Freytag & Hollensen (2001) referem as seguintes limitações:

- ✓ **Focalização nos números:** Por vezes as organizações focam-se nos dados e não nos processos que geram esses dados.
- ✓ **Falta de focalização nos clientes:** Devido aos baixos recursos que o *benchmarking* muitas vezes envolve, existe um alto grau de auto-avaliação que pode causar uma perda de focalização nos clientes.
- ✓ **Falta de focalização nos colaboradores:** As organizações que tentam produzir melhor *benchmarking* podem causar erros e cansaço aos seus colaboradores.
- ✓ **Dificuldade em obter informação útil dos concorrentes:** Concorrentes podem não cooperar. Reunir informação requer muito tempo, esforço e dinheiro.
- ✓ **Falhas na Implementação:** Por exemplo, quando os colaboradores não se envolvem no processo. Os colaboradores podem resistir à mudança. É necessário dar informação aos colaboradores de modo a melhorar os processos.
- ✓ **Processo contínuo:** O *benchmarking* é um processo contínuo, mas muitas organizações vêem-no como uma ferramenta casual.
- ✓ **Resistência ao conhecimento externo:** Muitas organizações consideram o que não é inventado por si inferior.
- ✓ **Exposição das fraquezas:** Algumas organizações não efectuem *benchmarking* por receio de revelarem fraquezas.
- ✓ **Âmbito das organizações estudadas:** Existe muitas vezes falha no âmbito das organizações estudadas.
- ✓ **Dificuldades Culturais ao transferir as “melhores práticas”:** Provenientes de culturas diferentes.

Verifica-se, portanto, que o *benchmarking* é uma ferramenta estruturada que deve ser utilizada no âmbito da gestão pela qualidade total, e que apresenta algumas vantagens mas também certas limitações.

2.1.6. Ferramentas da Qualidade

Desde há muito que as organizações utilizam ferramentas da Qualidade. A sua aplicação pode ser tanto para, identificação, análise ou resolução de problemas bem como para monitorização ou projectos de melhoria contínua.

Dentro das ferramentas da qualidade podemos ter ferramentas simples como por exemplo as 7 ferramentas básicas da qualidade (fluxograma, diagrama de Pareto, etc) ou ferramentas mais avançadas como, por exemplo, análise modal de falhas e efeitos (FMEA) ou a *Quality Function Deployment* (QFD).

No presente projecto interessa dar a conhecer e compreender o processamento de duas ferramentas da qualidade. Por um lado, a ferramenta 8D, que é utilizada em muitas reclamações efectuadas aos fornecedores sendo um componente essencial das reclamações, por outro a ferramenta A3 que é utilizada no projecto para auxílio na resolução do problema da gestão de reclamações. Tanto a ferramenta 8D como a A3 se baseiam em princípios de *Lean Production*, sendo ferramentas que podem ser utilizadas para melhoria de processos.

2.1.6.1. Ferramenta 8D

A ferramenta 8D é classificada como um MASP (método de análise e solução de problemas) (Campagnaro et al. 2008).

O mesmo autor refere que as MASP são ferramentas que normalmente se designam por yD em que y representa o número de passos. A ferramenta 8D, é constituída então por 8 passos.

Apesar de os passos variarem entre as diferentes ferramentas MASP, todas têm em comum um conjunto de passos que são (Campagnaro et al. (2008), Bouer(2005)):

- ✓ Identificação do problema.
- ✓ Acção de contenção.
- ✓ Identificação da causa raiz.
- ✓ Acção correctiva.
- ✓ Verificação da acção correctiva.
- ✓ Acção preventiva.

De acordo com Warren (2002) e Villiers (2008), a ferramenta 8D é um processo originário do programa, da FORD TOPS (Team Oriented Problem Solving), publicado em 1987. Villiers (2008) afirma que este método enfatiza o trabalho em equipa, tal como a descrição de TOPS pressupõe.

Garden (2007) refere que a ferramenta 8D fornece um processo lógico e simples para diferentes situações nomeadamente ao nível de: desenvolvimento e melhoria de produtos e processos, resolução de problemas e melhoria do serviço ao cliente. O método 8D é constituído por 8 passos (Garden (2007), Warren (2002), Bosch (2008), Villiers (2008)) descritos seguidamente:

D1 – Definição da Equipa – Deve ser estabelecido um líder que fica como dono do processo. O líder deve constituir uma equipa interdisciplinar de 4 a 6 pessoas com conhecimento na área em questão. Devem ser definidas regras e papéis para cada um.

D2 – Descrição do Problema – Descrição qualitativa e quantitativa do problema. Saber o quê? Como? Quando? Confirmar o problema e o seu âmbito.

D3 – Acção de Contenção – Iniciar a análise à causa raiz. Proteger os clientes do Problema. Tomar acções imediatas que impeçam lesão ao cliente. Informar os clientes sobre acções a ter, caso necessário.

D4 – Análise da Causa Raiz – Estabelecer a causa raiz de modo a eliminar o defeito de forma permanente. Alertar o cliente da quantidade prevista de problemas. Reproduzir o efeito para confirmar a causa. Nesta fase podem ser utilizadas ferramentas complementares como o diagrama causa – efeito, por exemplo.

D5 – Acções Correctivas – Definir acções para eliminar a causa do problema. Deve-se verificar a eficácia das acções em termos qualitativos e quantitativos.

D6 – Implementação das Acções Correctivas – Implementar as acções definidas no passo D5. Alertar o cliente para a data em que deixarão de existir problemas.

D7 – Acções Preventivas para Evitar Recorrência – Deve-se assegurar que as acções correctivas são devidamente implementadas e não são interrompidas e devem ser tomadas acções de forma a prevenir a recorrência do problema. Caso necessário deve-se actualizar a documentação relativa à Gestão da Qualidade para garantir que as alterações são mantidas (procedimentos, por exemplo). Se necessário deve ser dada formação aos colaboradores sobre alterações no processo ou produto.

D8 – Congratulação da Equipa – A solução é finalizada. Deve ser efectuada uma reunião para rever todos os passos anteriores.

Bosch (2008) enfatiza os seguintes pontos como principais num 8D: verificação e confirmação da causa raiz, eficiência das acções correctivas, implementação das acções correctivas, definição das acções preventivas.

Um exemplo de um template 8D pode ser consultado no presente projecto de investigação (ver anexo 1). No entanto, é de referir que não existe um template único, uma vez que cada organização pode desenvolver um consoante as suas necessidades.

2.1.6.2. Ferramenta A3

A ferramenta A3 consiste num método estruturado para a resolução de problemas desenvolvido pela Toyota (Sobek & Jimmerson (2004), Salvada (2008)). O nome dado a esta ferramenta advém de se utilizar um template numa folha de dimensão A3 (Harmon, 2008). Este mesmo autor diz que ao utilizar um A3 o gestor descreve um problema e propõe uma solução para o mesmo de forma sucinta, utilizando o documento para comunicar com a gestão de topo, que revê o documento e faz novas propostas se necessário. Tal como a ferramenta 8D, não existe um template universal existindo nuances consoante cada empresa e os seus objectivos. Salvada (2008) refere que esta ferramenta se baseia no ciclo PDCA (ver ponto 2.1.3. Norma ISO 9001:2008 e figura 9 e 11).

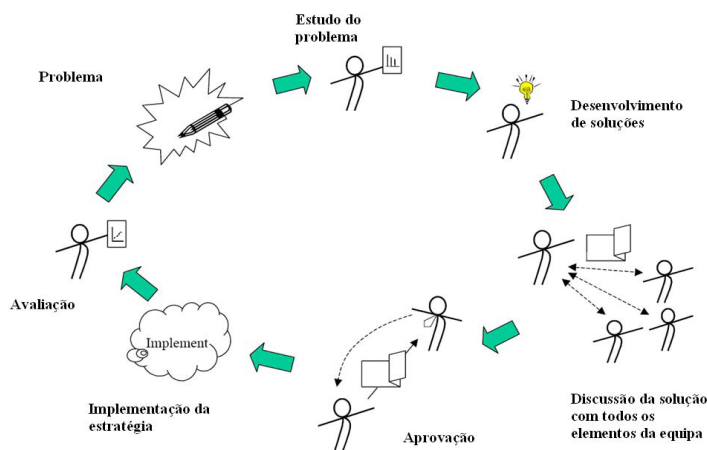


Figura 11. PDCA aplicado a A3 (Adaptado de Salvada (2008)).

Harmon (2008) refere que se pode complementar esta ferramenta com o uso de diagramas causa – efeito, fluxogramas, entre outras ferramentas.

Sobek & Jimmerson (2004), Harmon (2008) e Salvada (2008) apresentam textos semelhantes no que concerne à constituição de um A3. No entanto Sobek & Jimmerson (2004) efectuam uma abordagem mais completa, referindo que um A3 é constituído essencialmente por:

- ✓ **Título e Equipa** – Todos os A3's começam com um título, que tem que ser bastante descritivo do problema em causa. Deve focar o problema e uma possível solução. A seguir título o autor deve incluir alguma informação que seja necessária ao entendimento do problema (como foi descoberto, quais as partes envolvidas, por exemplo).
- ✓ **Estado Actual** – O autor deve efectuar um desenho do estado actual do problema / processo. Este é um dos pontos fundamentais do A3. Deve-se também quantificar o problema utilizando um indicador (% , quantidade). Os dados recolhidos para o desenho devem ser levantados através de observação directa, para evitar ambiguidades de relatos verbais. O diagrama desenhado permite a todos obter um entendimento do problema.
- ✓ **Análise das Causas Raiz** – Seguidamente o autor deve entender o porquê do problema, de forma a chegar às soluções, podendo fazê-lo de forma semelhante à de um 8D.
- ✓ **Estado Desejado** – Após conhecer o processo e as causas raiz é, então, fácil de efectuar o desenho do que se pretende obter. Este desenho é efectuado de forma análoga ao do estado actual.
- ✓ **Implementação** – Devem ser listados todos os passos a efectuar num tempo específico.
- ✓ **Monitorização** – O autor deve encontrar indicadores que provem que o estado actual do sistema actual é melhor que o anterior.
- ✓ **Resultados** – No final do relatório devem ser apresentados os principais resultados do A3 e, se necessário, efectuar novo A3.

Através do que é dito pelo autor verifica-se que esta ferramenta é essencialmente gráfica, partindo de esboços de situações concretas para chegar a possíveis soluções, o que pode permitir um maior envolvimento de todos nos projectos.

Sobek & Jimmerson (2004) afirmam que o A3 possui as vantagens de permitir documentar os processos directamente no local do trabalho, o que leva a um grau de detalhe que não é possível em salas de reunião, por exemplo; também o facto de serem utilizados desenhos em papel, que tornam os esquemas mais simples e perceptíveis, evitam a necessidade de formação para utilizar a ferramenta, uma vez que todos a compreendem.

2.2. Tecnologias e Sistemas de Informação

É pacífico afirmar que aos processos de negócio estão afectas muitas trocas de dados e/ou informação. Os exemplos destas trocas vão desde, encomendas a fornecedores, ao planeamento logístico, ao controlo de qualidade, etc. Quase todas as actividades organizacionais envolvem partilha de informação, e mesmo o próprio *benchmarking*, referido anteriormente também está fortemente associado com processos de partilha de dados e informação.

No paradigma que actualmente as organizações se enquadram, é muito importante ter os dados e/ou a informação no local certo, na quantidade exacta e no momento adequado, de forma a ajudar as empresas e serem mais competitivas.

Reflexo disto é a tentativa de redução, nas organizações, do *time-to-market* dos seus produtos e processos, o que implica um esforço de troca de informação a todos os níveis.

De acordo com Hernandez (1997) e Laudon & Laudon (2002) a informação consiste em dados que tenham sido processados de forma a terem significado, sendo úteis para quem os utiliza. Dados são factos em bruto que quando isolados não têm qualquer significado, como por exemplo, uma cor, um tamanho ou um diâmetro. Já quando se encontram contextualizados podem ter significado sendo nessa altura informação.

Juran & Godfrey (1998) consideram a medição e informação elementos base para a gestão pela qualidade total. No entanto, nem sempre os fluxos de informação e dados se apresentam optimizados, disponíveis e acessíveis. Desta forma, as Tecnologias de Informação e Comunicação (TIC) dão um contributo essencial para estas problemáticas, permitindo criar sistemas de suporte ao armazenamento de dados, distribuição e uso de informação (Beynon-Davies 2004). O mesmo autor afirma que as TIC modernas consistem em *hardware* (componente física, processadores, periféricos de entrada e saída), *software* (instruções para controle do *hardware*), estruturas de dados (estruturas de armazenamento de dados, discos rígidos) e tecnologias de comunicação (cabos, transmissores, routers). De acordo com Redman (1996) as TIC têm tido muito sucesso na automatização de operações fornecendo a plataforma ideal para o desenvolvimento de SI's. Em relação aos SI's, estes têm evoluído bastante nas últimas décadas, chegando a existir diferentes conceitos e tipologias de classificação de acordo com a visão de diferentes autores.

Laudon & Laudon (2002) define sistema de informação como um conjunto de componentes interrelacionados que reúnem, processam, armazenam e distribuem informação para suporte à tomada de decisão, coordenação e controlo numa organização. Podem também servir para auxiliar os gestores e colaboradores na análise e visualização de problemas e criação de novos produtos.

Rahman & Siddiqui (2006) referem que os SI's servem para medir, identificar, monitorizar e desenhar os atributos de produto e serviço que se traduzem em valor para os clientes.

Lagunas & Hellman (2007) afirmam que os SI's são uma forma de permitir a transferência de informação entre pessoas, transformando os dados em informação.

Já Beynon-Davies (2004) considera que os SI's fornecem informação para uma organização (ou parte dela) envolvendo comunicação entre pessoas e implicando recolha, processamento, distribuição e uso de informação.

Os SI's são bastante úteis na medida em que permitem automatizar certas tarefas e manipular dados de forma a convertê-los em informação útil. Malvius (2007) diz que a capacidade das organizações para adquirir e disseminar informação é um factor importante para a garantia da sua sobrevivência.

Segundo Laudon & Laudon (2002) são três as actividades básicas de um SI que participam na produção de informação de que uma organização necessita (figura 12):

- ✓ **Input** – recolhe os dados.
- ✓ **Processamento** – converte os dados em informação.
- ✓ **Output** – transfere a informação para as pessoas que dela necessitam.

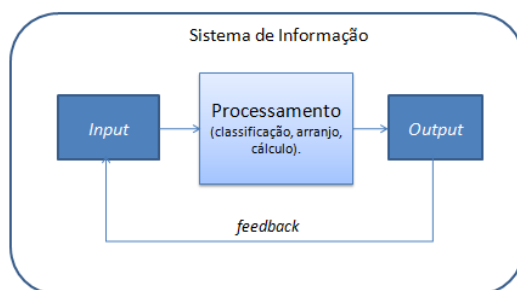


Figura 12. Actividades num Sistema de Informação (Adaptado de Laudon & Laudon (2002)).

Os autores referem também que pode existir um processo de *feedback* quando o *output* é utilizado no sentido de ajustar, corrigir ou avaliar o *input*.

Segundo Carriço (1996) existem dois tipos básicos de SI's (ver figura 13).

- ✓ **Sistemas de Processamento de Transacções** – têm o objectivo de registar e manter informação resultante da actividade desenvolvida pela organização.
- ✓ **Sistemas de Apoio à Decisão** – têm o objectivo de produzir e permitir o acesso a informação de apoio à tomada de decisão.

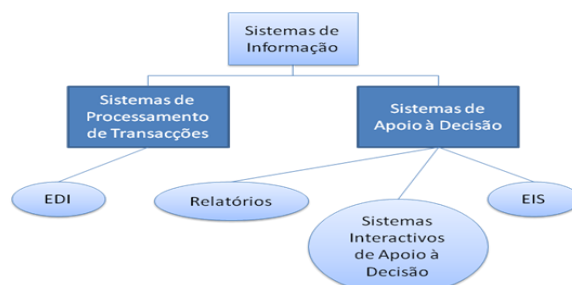


Figura 13.Tipos de SI's (Carriço (1996)).

Já Laudon & Laudon (2002) apresenta uma abordagem mais detalhada aos tipos de SI's existentes, em torno de seis categorias, como se pode ver na tabela 2.

Tabela 2. Tipos de Sistemas de Informação (Adaptado de Laudon & Laudon (2002)).

Tipo de Sistema	Input	Processamento	Output	Utilizadores
Sistemas de Suporte Executivo.	Dados agregados, internos e externos.	Gráficos, Simulação,	Projeções, Respostas a Consultas.	Gestores seniores.
Sistemas de Suporte à Decisão.	Pequenos ou grandes volumes de dados otimizados.	Simulações, Análises.	Relatórios Especiais, Respostas a Consultas.	Profissionais, Gestores.
Sistemas de Informação para a Gestão	Dados de transacções, grandes volumes de dados, modelos simples.	Relatórios de Rotina, Modelos Simples, Análises de Baixo Nível.	Relatórios e Resumos.	Gestores Intermédios.
Sistemas de Apoio ao Conhecimento.	Desenho de especificações.	Modelização, Simulação.	Modelos, Gráficos.	Profissionais, Técnicos
Sistemas de Escritório.	Documentos, Calendarizações.	Gestão de Documentação, Comunicação.	Documentos, Calendários, Emails.	Colaboradores.
Sistemas de Processamento de Transacções.	Transacções, Eventos.	Listagens, Actualizações, Ordenações.	Relatórios Detalhados, Listas, Resumos.	Pessoal Operacional, supervisores.

Independentemente do tipo de SI, a necessidade destes por parte das organizações é cada vez maior, não só pelo volume de dados que aquelas diariamente processam, como também pela rápida mutabilidade dos processos a que estão sujeitas.

2.2.1. Processo de Desenvolvimento de Sistemas de Informação

Os SI's têm um tempo de vida limitado o que contribui para que as organizações adquiram e desenvolvam aplicações de uma forma cada vez mais regular.

Lagunas & Hellman (2007) definem o processo de desenvolvimento de SI's como a construção e incorporação de SI's numa organização.

Como qualquer outra actividade em que se cria algo, é importante estabelecer e conhecer metodologias de desenvolvimento. Neste sentido, e particularmente no domínio dos SI's, vários autores apresentam algumas propostas associadas ao desenvolvimento deste tipo de produto.

Batini et al. (1992) descrevem o ciclo de vida de um SI através de um conjunto de passos, figura 14:

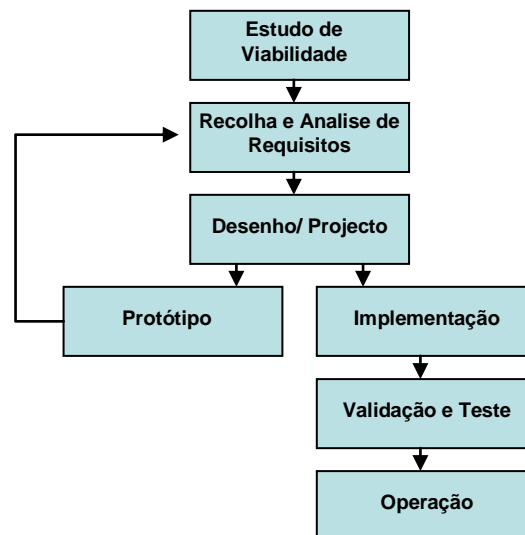


Figura 14. Ciclo de Vida dos SI's (Batini et al. (1992)).

- ✓ **Estudo de Viabilidade** – Consiste em determinar o custo da eficácia de várias alternativas no desenho do SI e as prioridades ao longo dos vários componentes do sistema;
- ✓ **Levantamento e Analise de Requisitos** – A recolha e análise referem-se à compreensão dos objectivos do SI, isto é, as aplicações, áreas e problemas que o sistema resolverá. Esta fase é focalizada também na interacção com os utilizadores na tentativa de compreender a suas necessidades;
- ✓ **Desenho** – O desenho é a fase em que se especifica a estrutura do SI. Inicialmente desenha-se a estrutura da base de dados e mais tarde das aplicações;
- ✓ **Protótipo** – Um Protótipo é uma versão simplificada do sistema para verificar na prática se as fases anteriores foram bem conduzidas.

O protótipo permite aos utilizadores verificar se o SI satisfaz as suas necessidades, sendo portanto muito utilizado para validar os requisitos;

- ✓ **Implementação** – A implementação é a fase relativa à programação da versão final do sistema;
- ✓ **Validação e Teste** – A validação e teste é o processo pelo qual se assegura que cada fase do processo de desenvolvimento foi cumprida. Verifica-se se a implementação reflecte as especificações do desenho;
- ✓ **Operação** – A operação começa com a introdução inicial de dados e termina quando o sistema se torna obsoleto. Durante a operação pode ser requerida manutenção, novas funções ou correcção de erros.

Já Davidson (2002) pelo seu lado, apresenta o modelo em espiral (figura 15) e cascata (figura 16) como sendo os mais utilizados no desenvolvimento de SI's.

Modelo em Espiral – Este modelo baseia-se em conceitos de prototipagem e implementação de sistemas evolutivos para se avaliarem custos e riscos. Este modelo divide as actividades de desenvolvimento em 4 quadrantes:

- i) Determinação dos objectivos do projecto – Verificar alternativas de construir ou comprar. Analisa-se se existe talento interno para o desenvolvimento, orçamento e tempo.
- ii) Avaliação das alternativas e identificação dos riscos – Escolha de alternativas para minimizar os riscos identificados.
- iii) Desenvolvimento de protótipo, baseado no trabalho resultante da fase anterior, e verificação da eficiência.
- iv) Reavaliação e planeamento do próximo ciclo.

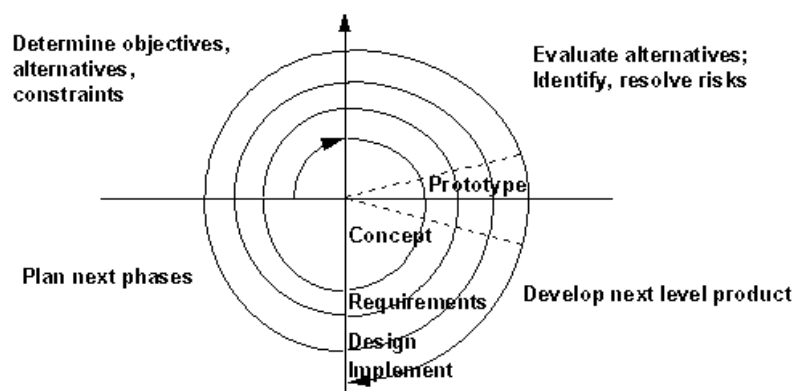


Figura 15. Modelo em Espiral (Davidson (2002)).

Neste modelo cada volta ao quadrante tem a designação de iteração. O modelo em espiral é semelhante ao apresentado por Batini et al. (1992), uma vez que também pressupõe iteratividade. De certa forma, apresenta alguma similaridade com o ciclo de Deming (abordado anteriormente) uma vez que se baseia em iterações de melhorias.

Quanto à iteratividade que caracteriza este modelo, Larman (2005) apresenta um conjunto de vantagens associadas:

- ✓ Menos falhas no projecto, melhor produtividade e menor taxas de defeito;
- ✓ Detecção precoce de riscos elevados (técnicos, requisitos, objectivos);
- ✓ Progresso visível desde cedo;
- ✓ *Feedback* mais precoce pela parte dos clientes o que pode levar a uma melhor adaptação ao sistema e levar a uma maior refinação dos requisitos;
- ✓ Menos complexidade da gestão, já que a equipa não depende de longas fases ou etapas complexas;
- ✓ Mais aprendizagem ao longo das iterações é evolutiva, podendo-se melhorar o desenvolvimento iteração por iteração.

Modelo em cascata – Este modelo é similar ao apresentado por Batini al. (1992), no que concerne às etapas envolvidas, sendo caracterizado pelos seguintes passos: definição de requisitos, desenho preliminar, desenho detalhado, implementação, teste unitário, teste de integração, teste do sistema, utilização do sistema, manutenção, não apresentando, no entanto, iteratividade. O modelo não apresenta iteratividade e pressupõe que cada fase deve estar desenvolvida a 100% antes de se passar para a seguinte (Davidson (2002), Larman (2005)), sendo por isso considerado muito restritivo (Davidson, 2002).

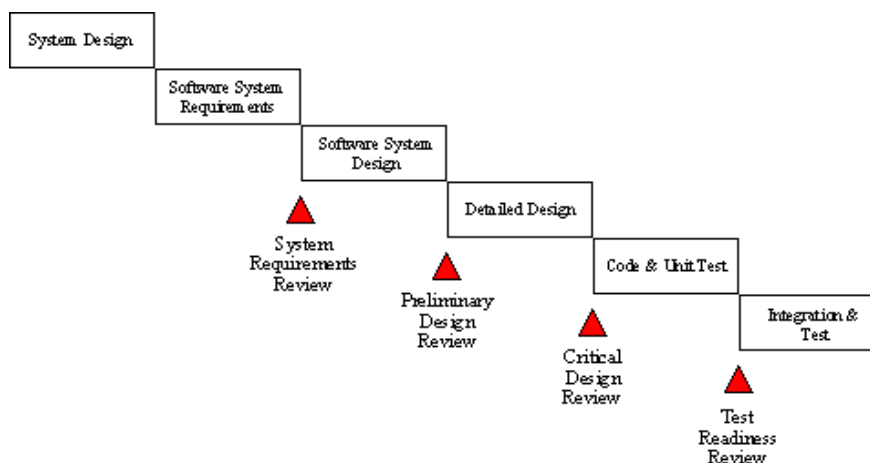


Figura 16. Modelo em Cascata (Davidson (2002)).

Larman (2005) refere que este tipo de modelo está associado a menor produtividade e maiores taxas de defeitos e falhas nos projectos, argumentando este facto com a possibilidade de existirem requisitos e especificações evolutivos que o modelo em cascata considera imutáveis e estáveis no tempo. Segundo o mesmo autor, a mudança é uma constante nos projectos de *software* e o *feedback* e adaptação são ingredientes chave para o sucesso.

No presente projecto será utilizada uma abordagem que se assemelha ao processo em espiral – o RUP (*Rational Unified Process*) - na medida em que enfatiza o conceito de Iteração (Booch et al.1999).

De acordo com Booch et al. (1999) o RUP não está associado a nenhum processo de desenvolvimento específico, baseando-se no entanto em três vectores principais ((Booch et al. (1999) & Jacobson e tal. (1999)):

- ✓ **Conduzido por casos de utilização** – os casos de utilização são utilizados como artefactos para estabelecer os comportamentos desejados do sistema, verificar e validar a arquitectura e comunicar entre os *stakeholders* no projecto. Os casos de utilização capturam requisitos funcionais e descrevem as funcionalidades do sistema.
- ✓ **Centrado na arquitectura** – utiliza a arquitectura do sistema como artefacto para construção, visualização e gestão do sistema a desenvolver, retratando a componente física.
- ✓ **Iterativo e incremental** – a abordagem iterativa e incremental enfatiza o trabalho por mini projectos. Em cada iteração os analistas identificam e especificam casos de utilização relevantes e criam um desenho recorrendo à arquitectura escolhida como guia. Devem-se seleccionar apenas as iterações necessárias para cumprir os objectivos de forma a economizar tempo e capital.

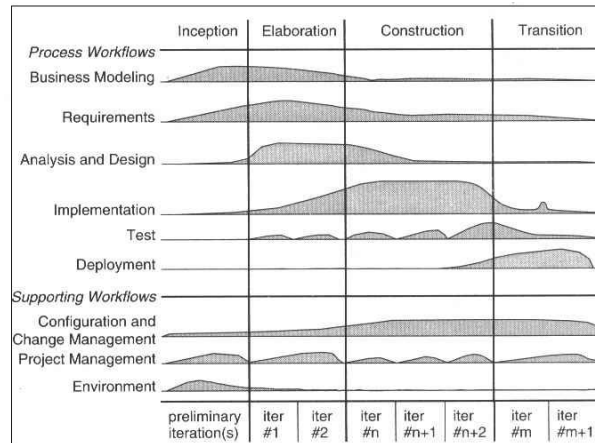


Figura 17. Processo Unificado de desenvolvimento (Booch et al. (1999)).

Na abordagem RUP (ver figura 17), o ciclo de desenvolvimento de *software* pode percorrer quatro fases (Booch et al. (1999) & Jacobson e tal. (1999)):

- ✓ **Início** – corresponde à fase em que existe uma ideia para o desenvolvimento. Esta ideia é suficientemente forte para prosseguir para a fase da elaboração.
- ✓ **Elaboração** – nesta fase a maioria dos casos de utilização são especificados e a arquitectura é desenhada. Os requisitos são prioritizados. No final o gestor de projecto deve saber quais as actividades a desenvolver e estimar os recursos para completar o projecto.
- ✓ **Construção** – nesta altura o produto é construído sendo a fase em que o sistema deve ficar pronto para transitar para os utilizadores. Nesta fase podem ser efectuadas pequenas alterações à arquitectura e aos requisitos. No fim deve conter todos os requisitos funcionais acordados.
- ✓ **Transição** – consiste no período em que o produto é lançado para os seus utilizadores. Normalmente é lançada uma versão experimental para avaliar possíveis deficiências que, caso existam, possam ser corrigidas. Esta fase envolve produção, formação dos utilizadores e disponibilização de sistemas de ajuda sob a forma de manuais.

Ao passar por estas quatro fases completa-se um ciclo de desenvolvimento podendo cada fase ser constituída por várias iterações (Booch, 1999). As fases são constituídas pelas actividades resumidas na tabela 3.

Tabela 3. Actividades no Processo Unificado de Desenvolvimento (Adaptado de Booch (1999))

Actividade	Descrição
Modelização do Negócio	Descreve a estrutura e dinâmica da organização.
Levantamento de Requisitos	Descreve o método, baseado em casos de utilização, para cumprimento de requisitos.
Análise e Desenho	Descreve as vistas arquitecturais. O que o sistema faz e como o faz.
Implementação	Tem em conta o desenvolvimento de software, teste unitário e integração.
Teste	Descreve os testes, procedimentos e métricas para localização de defeitos
Instalação	Contempla a entrega de uma versão operacional do sistema.
Gestão da Configuração	Controla alterações e mantém a integridade do projecto.
Gestão do Projecto	Descreve as diferentes estratégias de trabalhar com um processo iterativo.
Ambiente	Disponibiliza a infra-estrutura necessária para desenvolver o sistema.

2.2.2. Bases de Dados e Sistemas de Gestão de Bases de Dados

Muitas das actividades organizacionais têm associados tipos específicos de Sistemas de Informação, sendo as Base de Dados (BD's) um dos mais encontrados nas organizações.

De acordo com Carriço (1996) uma BD é um sistema com as finalidades de registar, actualizar, manter e disponibilizar informação relevante para as actividades organizacionais. Lucas et al. (2008) afirmam que uma BD é um conjunto inter-relacionado de dados de uma dada área.

Já para Hernandez (1997), e numa perspectiva de gestão, as BD's podem ser classificadas segundo duas vertentes:

- ✓ **Operacionais** - São utilizadas no dia-a-dia de uma organização. Este tipo de BD armazena dados dinâmicos, o que quer dizer que os dados podem mudar constantemente e reflectem a informação ao minuto (Ex. BD's de inventário).
- ✓ **Analíticas** - São utilizadas para armazenar e rastrear informação histórica e independente do tempo. Neste caso a BD é estática, o que significa que os dados nunca são, ou raramente são, modificados (Ex. BD's para dados estatísticos, projecções).

Um Sistema de Gestão de Base de Dados (SGBD) é uma aplicação que permite efectuar a gestão da base de dados, constituindo a interface entre os utilizadores e a estrutura implementada (Carriço 1996) e (Lucas et al. 2008).

Existem diferentes tipos de SGBD's, tais como: *IMS, DB2, Ingres, Informix Dynamic Server, Oracle Server, Sybase SQL Server, Microsoft SQL Server, Access*, (Lucas et al. 2008).

De acordo com Codd (1981) um SGBD tem que ter as seguintes capacidades: capacidade de armazenamento, pesquisa e actualização, catálogo para descrição dos dados, suporte de transacções, serviços de recuperação em caso de falhas, serviços de autorização de forma a garantir o acesso aos utilizadores certos, suporte para comunicação.

Os SGBD's permitem implementar modelos de dados, ao mesmo tempo que oferecem aos utilizadores a possibilidade de manipularem aqueles através de interfaces gráficas, sem terem a necessidade de compreender a estrutura da BD's.

No entanto, para implementar os modelos dados é necessário primeiro encontrar o respectivo modelo conceptual, usando para tal uma qualquer abordagem de desenvolvimento, ao longo de um processo conhecido por Modelização ou Projecto.

2.2.3. Projecto de Bases de Dados

A fase de projecto, também conhecida pela fase de desenho ou modelização, é bastante importante no desenvolvimento de uma BD, sendo também um processo revestido de alguma complexidade (Hernandez 1997). Batini et al. (1992) afirmam que para além de complexo, envolve várias decisões, devendo-se por isso decompor o problema em sub-problemas, procurando posteriormente a sua resolução de forma independentemente.

Para esta fase, Hernandez (1997) defende uma metodologia composta por um conjunto de etapas:

- i) **Definição da Missão e Objectivos para a BD** – A missão define o propósito da base de dados e os objectivos definem as tarefas que os utilizadores farão com os dados no respectivo sistema.
- ii) **Análise da Base de Dados Actual** – Verificação dos requisitos da organização recorrendo a BD's existentes e entrevistas com os utilizadores e gestores, de forma a determinar como usam BD's no seu quotidiano.

- iii) **Criação das Estruturas de dados** – Construção das tabelas através da identificação dos assuntos que serão abordados na BD. Escolha os campos que melhor caracterizam esses assuntos, e definir as chaves primárias.
- iv) **Determinação e Estabelecimento das relações entre as tabelas** – Identificação das relações entre as tabelas através da definição de chaves primárias e chaves estrangeiras.
- v) **Determinação e Definição das Regras do Negócio** – Realização de entrevistas para identificar restrições nos dados baseadas na forma como a organização vê e utiliza os dados. Estas restrições são consideradas como regras de negócio que impõem várias restrições aos dados.
- vi) **Determinação e Construção de Consultas** – Levantamento, junto dos utilizadores, das várias formas como vêm os dados na BD. Após as várias perspectivas terem sido identificadas, estas devem ser integradas na BD.
- vii) **Revisão da Integridade dos Dados** – Revisão da especificação, tabelas encontradas e campos definidos, de forma a assegurar que todos os critérios definidos para o projecto tenham sido considerados. Por último, deve testar-se a validade de cada relação e confirmar as regras de negócio.

Este autor refere-se ao desenho da BD de uma forma muito prática, no entanto não enfatiza a necessidade de existência de modelos de abstracção e desenho. Estes modelos fornecem uma base para construção de BD's propriamente ditas.

Batini et al. (1992) pelo seu lado e com uma visão mais estratégica, apresenta um modelo mais geral, dividindo o processo de desenho em 3 sub-processos:

- i) **Desenho conceptual** - Começa com a especificação dos requisitos num esquema conceptual, sendo este uma descrição de alto nível da estrutura da BD com o propósito de descrever o tipo de dados que irá suportar.
- ii) **Desenho lógico** - O desenho lógico inicia-se a partir do esquema conceptual e resulta num esquema lógico. O esquema lógico é uma descrição da estrutura da BD que pode ser processada num SGBD. Um modelo lógico é uma linguagem que pode ser utilizada para especificar esquemas lógicos, tendo como exemplos os modelos relacionais, em rede e hierárquicos.
- iii) **Desenho Físico** - O desenho físico é efectuado através do esquema lógico e resulta no esquema que descreve as estruturas de armazenamento e métodos de acesso utilizados para aceder aos dados.

2.2.4. Modelos de Bases de Dados

Existem diversos modelos de BD's, tais como modelo em Rede, modelo Hierárquico, modelo Relacional, modelo Orientado a Objectos (Hernandez, 1997), sendo neste projecto adoptado o modelo relacional.

2.2.4.1. *Modelo Relacional*

O modelo de BD relacional surgiu em 1970 pela mão de Codd (Hernandez, 1997). Neste modelo os dados são armazenados em tabelas (figura 17), ligadas entre si através de relações de diversos tipos. Codd (1979) diz que uma tabela consiste num conjunto de *tuplos* (linhas), sendo cada um constituído pelo mesmo conjunto de atributos (colunas).

De acordo com Davies-Beynon (2004) uma tabela obedece ao seguinte conjunto de regras:

- ✓ Cada tabela na BD tem que ter um nome distinto.
- ✓ Cada coluna tem que ter um nome distinto da tabela.
- ✓ Todas as entradas numa tabela têm que ser do mesmo tipo, sendo definidas no mesmo domínio.
- ✓ A ordem das colunas é indiferente.
- ✓ Cada linha numa relação tem que ser distinta. Linhas duplicadas não são permitidas.
- ✓ A ordem das linhas é indiferente.
- ✓ Cada célula deve conter apenas um valor atómico.

Desta forma este modelo supera algumas dificuldades tidas com o modelo hierárquico e o em rede, uma vez que era necessário um conhecimento das estruturas para funcionar com a BD.

Codd (1979) descreve o modelo relacional como um conjunto de tabelas que se regem por um conjunto de regras de inserção (Insert), actualização (Update) e eliminação (Delete) segundo um conjunto de princípios assentes na Álgebra relacional.

O mesmo autor refere noutra obra que o modelo relacional é constituído por três componentes (Codd, 1981):

- ✓ Estrutura de dados que retrata os blocos de construção – tabelas, atributos, chaves primárias, *tuplos*, etc.
- ✓ Colecção de operadores e regras que podem ser aplicados para manipular dados – *select*, *join*, etc.
- ✓ Regras de integridade que definem a consistência da BD.

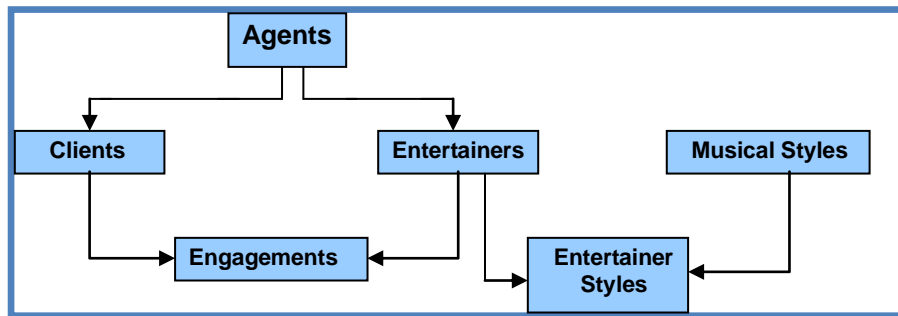


Figura 18. Modelo Relacional (Hernandez (1997)).

Duas tabelas estão implicitamente ligadas pelo facto de possuírem um campo comum, sendo que numa tabela esse campo é chave primária, e na outra constitui chave estrangeira. Por exemplo, a tabela *Agents* está ligada à tabela *Clients* porque possuem o campo *AgentID* em comum (ver figura 19).

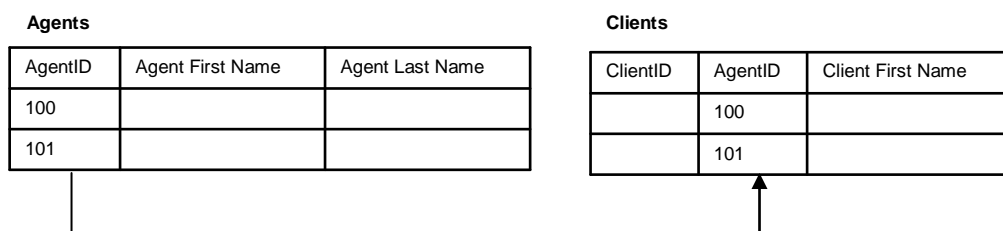


Figura 19. Exemplo Ligação em Modelo Relacional (Hernandez (1997)).

Desde que o utilizador conheça as relações entre as tabelas pode aceder aos dados de uma forma variada.

Codd (1981) diz que todas as inserções, actualizações ou eliminações de relações têm que seguir as seguintes regras:

- ✓ **Integridade da Entidade** – nenhuma chave primária de uma relação pode ser nula ou ter um componente nulo.
- ✓ **Integridade Referencial** – O valor de uma chave estrangeira ou é nulo, ou contém um valor que é chave primária na entidade de onde foi importada.

Hernandez (1997) e Codd (1970) apresentam algumas vantagens do modelo relacional:

- ✓ Integridade dos dados em vários níveis (campos, tabelas, relações).
- ✓ Dados facilmente consultados.
- ✓ Garantia de consistência dos dados.
- ✓ Independência lógica e física de aplicações de BD's.

Hernandez (1997) refere que uma das desvantagens deste tipo de modelo é que os programas de *software*, que se baseiam nele, são muito lentos.

Para além das Tabelas, as BD's relacionais contemplam um conjunto de relações.

✓ Relações

As relações são ligações estabelecidas entre tabelas. Cada relação pode ser caracterizada pelo grau e tipo de participação. Quanto ao grau de participação, que corresponde ao mínimo e máximo número de registos numa tabela que podem-se relacionar com um único registo da outra tabela, podemos encontrar relações de grau 'um-para-um', de 'um-para-muitos' ou então de 'muitos'-para-muitos':

i) Relações de Um-para-Um

Acontecem quando, em duas tabelas, um registo de uma está relacionado apenas com um registo da outra, e vice-versa. No exemplo retratado na figura 20, apenas um registo da tabela A está para um da tabela B e vice-versa.

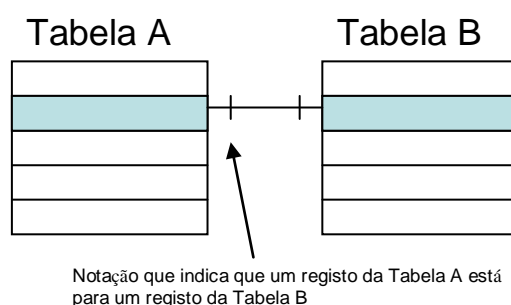


Figura 20. Exemplo Relação de Um-para-Um (Hernandez (1997)).

ii) Relações de Um-para-Muitos

Este tipo de relações existe quando um registo da primeira tabela se relaciona com um ou mais registos da segunda tabela. Mas um registo na segunda tabela pode apenas relacionar-se com um registo na primeira (ver figura 21).

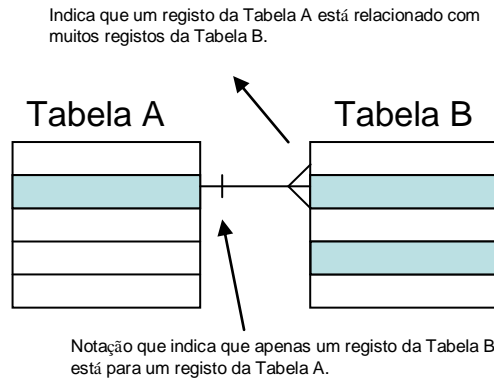


Figura 21. Exemplo Relação de Um-para-Um (Hernandez (1997)).

iii) Relações de Muitos-para-Muitos

As relações de muitos-para-muitos existem entre um par de tabelas se um registo da primeira tabela está relacionado com um ou mais registos da segunda, e um registo da segunda tabela está relacionado com um ou mais da primeira (ver figura 22).

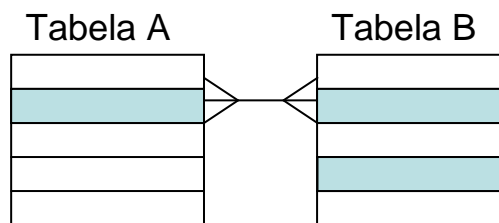


Figura 22. Exemplo Relação de Muitos-para-Muitos (Hernandez (1997)).

Ainda que a descrição feita anteriormente se tenha baseado num tipo específico da notação - a Entidade-Relação (ER) - existem outros tipos, que induzem a construção de modelos de acordo com outras notações, tais como a OMT e a UML (ver tabela 4)

Tabela 4. Notações Comuns para Relações (Adaptado de Martin & Odell (1998)).

	Cada A está apenas para um B.	Cada A está para um ou muitos de B	Cada A tem que estar para zero ou um de B	Cada A pode estar para qualquer número de B's.
UML				
Martin / Odell (1ª edição)				
Booch (2ª edição)				
Coad / Yourdon				
Jacobson (unidireccional)				
OMT				

Quanto ao tipo de participação, existem 2 tipos possíveis no âmbito de uma relação: obrigatória ou opcional. Exemplificando com uma relação entre 2 tabelas designadas por Tabela A e Tabela B: se os registos da Tabela A têm que existir antes de se poder inserir registos na Tabela B, a participação da Tabela A é obrigatória; no entanto se não existir essa necessidade a sua participação é opcional.

Embora a BD do presente projecto seja do tipo relacional, o modelo conceptual que esteve na sua origem assentou num paradigma orientado a objectos, utilizando a linguagem UML.

2.2.4.2. Modelo Orientado a Objectos

O modelo Orientado a Objectos, como o nome indica, baseia-se em objectos. Martin & Odell (1998) definem objecto como qualquer coisa a que um conceito se aplique.

Larman (2005) diz que na análise orientada a objectos existe ênfase em encontrar objectos no domínio do problema, dando o exemplo de um SI de voo em que alguns dos conceitos incluem avião, voo e piloto. A orientação a objectos baseia-se no conceito de objectos do mundo real, podendo-se esses ser caracterizados por um conjunto de atributos e operações (Martin & Odell, 1998).

Os mesmos autores referem que esta orientação fornece uma forma de organização do conhecimento conceptual seja esse conhecimento expresso em forma de regras, lógica, funções, linguagem relacional, ou qualquer outra coisa.

Larman (2005) refere que no desenho orientado a objectos põe ênfase na definição dos objectos de software e na forma como colaboram para preencher os requisitos. A UML é uma linguagem que surgiu nesse paradigma, como brevemente se explica a seguir.

2.2.5. Modelização em UML

Quando se desenvolve um sistema é importante projectá-lo antes de proceder à sua construção física. A modelização permite efectuar uma serie de cenários possíveis de forma rápida e simples sem recorrer a soluções dispendiosas e inflexíveis como sejam o desenvolvimento dos sistemas físicos por tentativas.

Sinfic (2005), diz que modelizar ajuda a reduzir o risco associado ao desenvolvimento de sistemas, permitindo às organizações especificar e comunicar informação das suas aplicações em diversas perspectivas para os diferentes intervenientes envolvidos no desenvolvimento do sistema.

Cunha (2001) diz que através da modelização se podem ter algumas vantagens, como:

- ✓ Auxiliar a visualização do sistema como ele é ou como queremos que seja.
- ✓ Auxiliar a especificação e estruturação do sistema.
- ✓ Permitir a definição de um formato que serve de guia para a construção do sistema.
- ✓ Permitir efectuar documentação.

Por sua vez, a modelização visual, baseada em componentes gráficas, para além das vantagens supracitadas, permite uma mais fácil comunicação com o cliente do sistema. Através desse tipo de modelização, é possível desenhar o sistema, criar alternativas, funcionalidades e funções de forma rápida e fácil, sem ter que enfrentar os custos da implementação.

A Linguagem UML potencia a modelização visual, sendo essa uma das razões que contribuiu para a sua popularidade. É uma linguagem gráfica de visualização, especificação, construção e documentação de artefactos de sistemas de software (Booch, 1999; Aalst & Hee, 2002; Cunha, 2001; e Larman, 2005). Trata-se de uma linguagem de simples interpretação, com forte componente visual e padronizada (Aalst & Hee, 2002; Sinfic 2008).

Com a UML, podemos comunicar os conceitos a terceiros sem ambiguidade, compreender situações que seriam difíceis por análise directa do código e obter maior facilidade na comunicação (Cunha (2001).

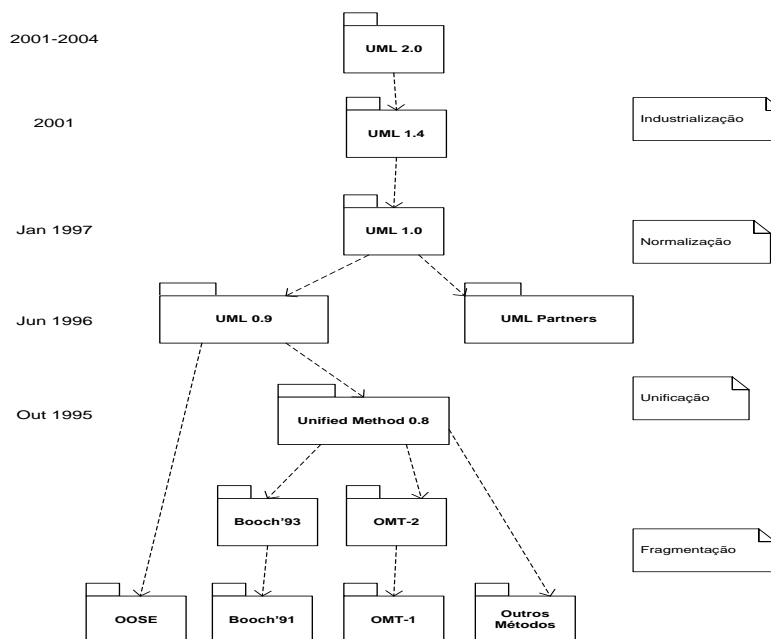


Figura 23. Evolução da UML (Adaptado de Aguiar et al. (2007)).

O UML surgiu em 1994 (ver figura 23) quando James Rumbough se juntou a Grady Booch na Rational Software Corporation. Ambos tinham trabalhado em métodos orientados a objectos de nome OMT (*Object Modeling Technique*). Mais tarde, em 1995, juntou-se um terceiro elemento Ivar Jacobson (Aalst & Hee (2002), Larman (2005), Martin & Odell (1998)).

Na realidade a linguagem UML veio responder à necessidade de existência de uma linguagem visual consistente, tendo o seu marco histórico sido registado em 1997, altura em que foi lançada a primeira versão da UML (UML 1.0) (Jacobson et al., 1999).

Ao longo dos anos foram surgindo novas versões, tendo até mais recentemente, com a versão UML 2.0, surgindo novos diagramas (ver figura 23).

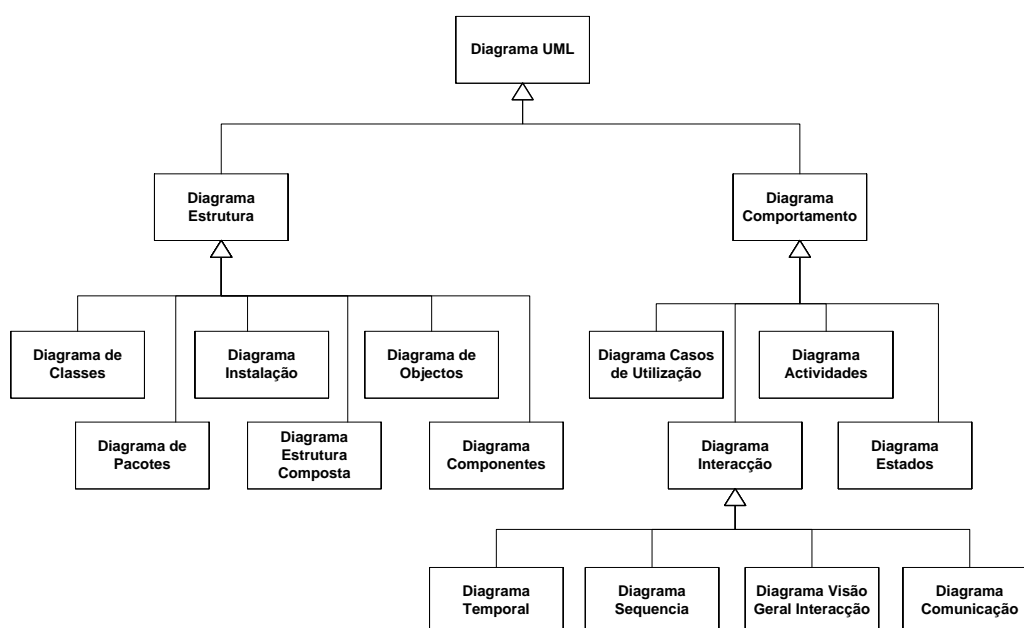


Figura 24. Hierarquia dos Diagramas UML (Aguiar et al. (2007)).

Actualmente contam-se já 13 tipos de diagramas divididos em 3 categorias (Tim, 2003, Aguilar et al., 2007) (ver figura 24), podendo-se encontrar uma breve descrição daqueles no Anexo 2.

2.2.5.1. Formas e Perspectivas de Utilização da UML

Em relação às formas de utilização da linguagem UML, Larman (2005) distingue três formas de aplicação, que vão desde técnicas informais de representação do domínio do problema, até a uma especificação total do sistema pronta a implementar.

- ✓ **UML como Esboço** – em que são feitos diagramas informais e incompletos (muitas vezes feitos manualmente) criados para explorar dificuldades no problema ou solução, maximizando assim o poder das linguagens gráficas.
- ✓ **UML como *Blueprint*** – em que são efectuados desenhos relativamente detalhados utilizados para compreender melhor o código existente nos diagramas ou mesmo para gerar código.
- ✓ **UML como Linguagem de Programação** – em que é efectuada uma especificação completa e executável de um sistema de software, podendo o próprio código ser gerado automaticamente.

O mesmo autor refere ainda que a UML pode ser utilizada em três perspectivas e tipos de modelos diferentes, que são:

- ✓ **Perspectiva Conceptual** – os diagramas descrevem coisas do mundo real ou do domínio de interesse.
- ✓ **Perspectivas de Especificação** – os diagramas descrevem abstracções de software ou componentes com especificações e interfaces mas sem o comprometimento de uma implementação particular (C# ou Java).
- ✓ **Perspectivas de Implementação** – os diagramas descrevem implementações de software particulares numa tecnologia específica (como Java).

2.2.6. Desenho de Interface do Utilizador

O desenho de Interface é um passo muito importante quando se desenvolve um SI. É através deste que os utilizadores vêem e executam todas as funcionalidades do Sistema. Como tal, torna-se de extrema importância a construção correcta de interfaces, utilizando como suporte de desenvolvimento as metodologias mais adequadas.

Uma boa interface facilita a navegação e a inserção de dados, tornando-a mais intuitiva para o utilizador e com uma menor probabilidade de introdução de erros.

“Ao contrário dos computadores o homem erra facilmente.”
(Constantine & Lockwood, 1999)

Shneiderman & Plaisant (2005) têm contribuído para a investigação nesta área, tendo proposto uma técnica assente em três pilares para a construção correcta de interfaces (ver figura 25).

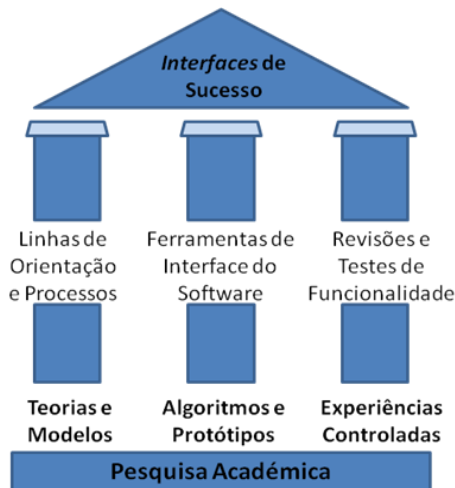


Figura 25. Três pilares para a criação de bons interfaces (Adaptado de Shneiderman & Plaisant (2005)).

i) **Linhas de Orientação e Modelos** – Desde cedo no processo de desenho se deve obter documentação com linhas de orientação de forma a ser garantida uma padronização com as restantes aplicações existentes, ou apenas para satisfazer o desejo do cliente. Apesar das nuances existentes para os diferentes projectos as linhas de orientação devem ser consideradas:

- ✓ **Palavras, ícones e gráficos** - Terminologia, abreviaturas, tipo de letra tamanhos de letra, estilos, ícones, botões, gráficos, grossuras de linha, cores e fundos.
- ✓ **Layout do ecrã** - Selecção de menus, preenchimento de formulários e caixas de diálogo, *Feedback* de mensagens, justificação, margens, entrada de dados, formatos de *display* para os itens e listas, cabeçalhos e Rodapés.
- ✓ **Input e Output** - Teclado, *display*, cursor, Sons, vozes de *feedback*, *inputs* especiais, tempo de resposta às tarefas).
- ✓ **Sequências e Acções** - Manipulação – clique, arrastamento – comandos, sequências, atalhos e funções programadas.
- ✓ **Ajuda** - Manuais de Ajuda.

ii) **Ferramentas de Interface do Software** – Uma das dificuldades no desenvolvimento de Interfaces consistem na inexistência de uma ideia do aspecto final do sistema pelos utilizadores e clientes. De forma a colmatar esta lacuna, uma vez que é difícil e dispendioso de alterar sistemas já implementados, devem ser utilizados protótipos dos menus. Os protótipos podem ser desenvolvidos de forma simples e sem que exista processamento, uma vez que servem apenas para assegurar aspectos estéticos.

iii) Experiências Controladas – Devem ser efectuadas experiências à funcionalidade da Interface. Para isso podem ser feitas análises por peritos, testes com os utilizadores ou testes com recurso a ferramentas automáticas de análise. Os procedimentos utilizados variam consoante os objectivos do estudo, o número de utilizadores, o perigo dos erros e do orçamento existente.

Apesar dos três pilares apresentados por Shneiderman & Plaisant (2005) serem bons indicadores para o desenvolvimento de interfaces, não constituem uma metodologia efectiva. As metodologias podem ser úteis para se conseguir cumprir os projectos atempadamente e também porque permitem uma abordagem aos problemas com ferramentas já testadas, sem que isso constitua um custo extra. Yang et al. (1998) apresentam uma série de metodologias de apoio ao desenvolvimento de interfaces entre as quais:

- ✓ ***L.U.C.I.D (Logical User Centered Design)*** – avalia as necessidades dos utilizadores e foca-se em satisfazê-las de forma simples sem preocupação com a componente técnica;
- ✓ ***Music (Measuring Usability of Systems in Context)*** – especifica objectivos de funcionalidade e avalia se os mesmos foram alcançados;
- ✓ ***Participory Design*** – foca-se na participação dos utilizadores no desenvolvimento;
- ✓ ***Contextual Design*** – Utiliza dados recolhidos para a definição do produto / processo numa abordagem centrada no cliente;
- ✓ ***Performance Centered Design*** – foca-se no alcance de performance do sistema.
- ✓ ***The Scenario-Based Engineering Process*** – baseia-se em cenários para efectuar análise, desenvolvimento e testes;
- ✓ ***Discount Usability Engineering*** – baseia-se em cenários, pensamento simples e avaliação heurística;
- ✓ ***Usability Methods*** – centra-se nos utilizadores e nas suas necessidades.

Shneiderman & Plaisant (2005) afirmam que uma das metodologias mais testada e utilizada é a *L.U.C.I.D*. Este modelo baseia-se na prototipagem, testes funcionais e no conceito de iteratividade.

No presente projecto, para além do aspecto gráfico e da forma como as funcionalidades se apresentam, a questão da interface foi também trabalhada ao nível da navegação, tendo o Visual Basic um grande papel nessa concretização (Anexo 10).

3. Desenvolvimento de um Sistema de Gestão de Reclamações: *Bosch Termotecnologia S.A*

Este capítulo será dedicado à descrição da componente prática deste projecto, mais especificamente ao processo de análise, modelização e implementação de um protótipo de um Sistema de Gestão de Reclamações na *Bosch Termotecnologia SA*. Em primeiro lugar faz-se referência ao processo de negócio antes da existência do sistema, compreendendo-se assim algumas das lacunas existentes e todo o processo afecto às reclamações, tanto ao nível de fluxos como partes envolvidas.

Seguidamente é apresentado o processo e o levantamento de requisitos, recorrendo para isso ao método de resolução de problemas A3 e aos resultados dos vários estudos efectuados, nomeadamente o *Benchmarking* e comparação de sistemas existentes no mercado. Posteriormente apresenta-se o processo de negócio, admitindo a existência do sistema, bem como o diagrama de casos de utilização que suportará as interacções.

Apresenta-se ainda a estrutura estática do sistema através do diagrama de classes, sendo exemplificado com base num diagrama de sequência uma interacção envolvendo objectos externos (actores do diagrama de casos de utilização) e objectos do sistema (classes do diagrama de classes).

Por último, é abordada a transição da modelização em UML para o modelo relacional e construído o protótipo do Sistema

A abordagem utilizada para o desenvolvimento prático é a apresentada por Batini et al. (1992) e Davidson (2002) sendo compaginada com o desenvolvimento em espiral, e com o processo iterativo baseado na prototipagem.

3.1. Análise do Processo Existente

Quando se desenvolve um SI têm que se conhecer os processos de negócio afectos ao mesmo. Não basta só efectuar o levantamento dos requisitos; devendo-se saber o estado actual da situação, para compreender melhor o que se pretende no futuro.

Nesse sentido, e neste projecto em particular, foi importante compreender o fluxo de actividades que representa o processo de reclamação, sendo possível descrever esse processo da seguinte forma:

1. Surge um problema que gera uma reclamação, feita pela Origem, para a Qualidade das Compras.

2. A reclamação é analisada pelo responsável do componente em causa, que pode ou não validar a reclamação, gerando a situação a ou b.
 - a. Se a reclamação for válida, é analisada a gravidade do problema para avaliar se existe necessidade de pedir ou não um 8D, gerando a situação a.1. ou a.2.
 - a.1. Se for necessário pedir um 8D, é enviado um template ao fornecedor, passando-se para o passo a.1.1.
 - a.1.1. O fornecedor preenche e seguidamente envia o template, passando-se para o ponto a.1.1.1.
 - a.1.1.1. O responsável valida o 8D podendo ocorrer duas situações, a.1.1.1.1 ou a.1.1.1.2.
 - a.1.1.1.1. Se o 8D não for válido volta para o ponto a.1.
 - a.1.1.1.2. Se o 8D for válido vai para o ponto 3.
 - a.2. Se não for necessário pedir um 8D, o responsável pede acções normalmente.
 - a.2.1. O fornecedor responde às acções pedidas passando-se para o ponto a.2.2.1
 - a.2.2.1. O responsável valida as acções podendo ocorrer duas situações, a.2.2.1.1 ou a.2.2.1.2.
 - a.2.2.1.1. Se as acções não forem válidas, volta para o ponto a.2.
 - a.2.2.1.2. Se as acções forem válidas, vai para o ponto 3.
 - b. Se a reclamação não for válida, é dado o *feedback* à origem e termina o processo.
3. A origem verifica os resultados podendo ocorrer duas situações, 3.a. ou 3.b.
 - 3.a. Se existir ausência de defeito para o processo.
 - 3.b. Se existir defeito volta-se ao ponto 1.

Na figura 26 podemos ver o mesmo processo representado de forma gráfica com recurso ao diagrama de actividades da notação UML, em que as actividades são representadas por elipses e as decisões por losangos.

Os diagramas de actividades servem para representar processos de negócio, comportamento de algoritmos e detalhar casos de utilização, sendo inspirados nos fluxogramas em que se podem representar acções e actividades.

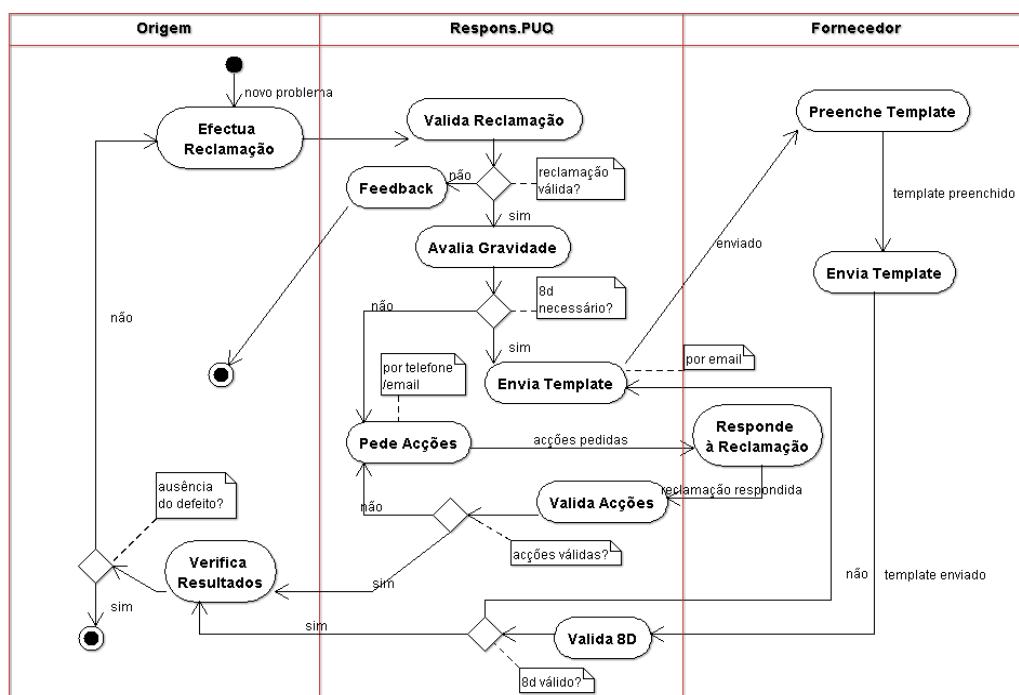


Figura 26. Diagrama de Actividades – Antes do sistema.

Cada actividade só pode ser desempenhada por um dos actores e, como tal, só pertence a uma pista.

3.2. Análise de Requisitos

Após a compreensão do problema e do processo associado ao mesmo, foi feita a análise de requisitos recorrendo a várias técnicas: a ferramenta da qualidade A3 (que resulta de reuniões com o grupo de Engenharia) e estudos de *benchmarking* e sistemas existentes no mercado, foram as técnicas que mais se evidenciaram neste.

3.2.1. Ferramenta A3

Utilizando a ferramenta A3 (descrita no ponto 2.1.6.2) efectuou-se uma abordagem estruturada ao problema, tendo-se feito uma análise em equipa ao mesmo, de acordo com o modelo de Sobek & Jimmerson (2004), que se compagina com o *template* utilizado pela empresa.

O A3 é efectuado em inglês uma vez que esta é a língua oficial da Organização (ver anexo 3). Através desta ferramenta conseguem-se levantar alguns dos requisitos do sistema.

i) Título e Equipa - Complaint Management System.

O título, tal como referido pelos autores, aborda o problema e a solução numa frase curta. A equipa envolvida é constituída por todos os colaboradores do departamento PUQ.

ii) Estado Actual

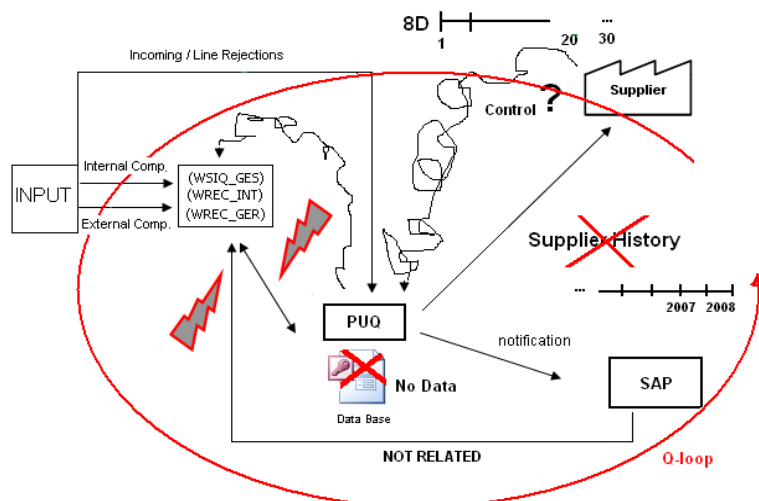


Figura 27. Estado Actual – sem sistema.

Como foi referido pelos autores da ferramenta o estado actual retrata o problema existente na forma de um desenho. Neste caso o desenho demonstra, de uma forma gráfica e simples, os problemas existentes nos fluxos de informação (ver figura 27). Por exemplo, a não existência de uma BD que centralize a informação proveniente das origens não permite aos Eng.ºs de Qualidade efectuar eficientemente uma gestão das reclamações aos fornecedores. Estes, por sua vez, são difíceis de controlar e gerir no que respeita ao seu *feedback*, uma vez que não existe uma metodologia para o fazer. O facto da informação dos fornecedores ser enviada por correio electrónico e telefone, torna difícil a sua conversão para análises e histórico.

iii) Análise das Causas Raíz

Após reunião com os restantes colaboradores chegaram-se às seguintes causas da não adequada gestão das reclamações:

- ✓ Não existência de histórico de reclamações.
- ✓ Desconhecimento das relações entre o SAP e as reclamações efectuadas, para os fornecedores do tipo C.

- ✓ Controlo dos Tempos de 8D's efectuado manualmente.
- ✓ Difícil de rastrear problemas redundantes.
- ✓ Ausência de padronização no tratamento das reclamações.
- ✓ Círculo da Qualidade aberto para os fornecedores do tipo C.

O sistema a ser desenvolvido tem que ter em conta estas causas.

iv) Estado Desejado

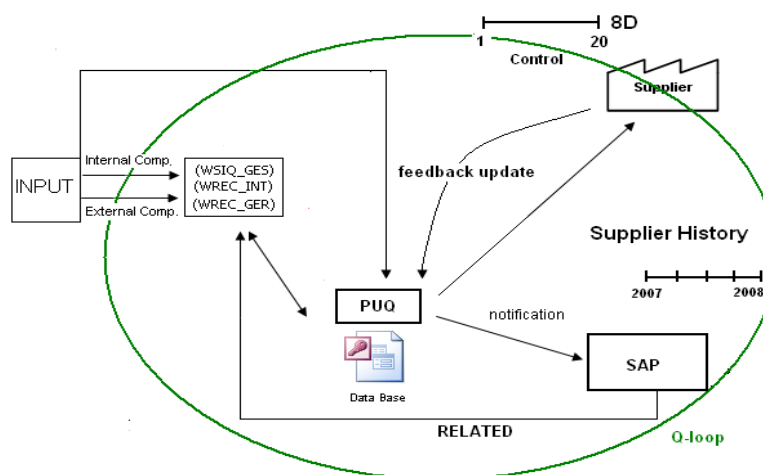


Figura 28. Estado desejado – futuro.

Como foi referido pelos autores da ferramenta, o estado desejado retrata o que se pretende no futuro (ver figura 28). Neste caso a existência do sistema permite otimizar os fluxos de informação entre o Cliente interno / PUQ / Fornecedor. Com a existência de uma BD é possível conectar as 3 partes envolvidas, automatizando tarefas e evitando assim a manipulação de ficheiros dispersos entre as partes.

v) Implementação

A implementação a efectuar tem em conta um conjunto de acções que devem ser levadas a cabo. Através da representação da figura 29 podem-se analisar as principais acções efectuadas e a sua calendarização.

O *status* retrata o estado em que a acção se encontra; o *effect* se a acção teve o efeito desejado ou não. Podemos verificar que a acção *Study of the system requirements* tem um *status* bom e um efeito bom. Esta classificação acontece porque, uma acção pode ter sido bem desenvolvida, sem no entanto, criar um efeito favorável (positivo).

	No:	Implementation actions	By whom	Time scale (Months)										Status	Effect
				1	2	3	4	5	6	7	8	9	10		
Action plan		Integration in PUQ team	Rui A.											G	G
		Study of the system requirements	Rui A.											G	G
		Bosch Security System Visit	Rui A.											G	G
		Development of Pilot System	Rui A.											G	G
		Development of Final System	Rui A.											G	G
		Implementation	Rui A.											Y	Y
		Benchmarking	Rui A.											Y	Y
Key (Status & Effect):				Good: G		Bad: R		Okay: Y							

Figura 29. Acções para implementação.

Existem duas acções que se encontram com o *status* e *effect* no *Okay*, uma vez que a sua implementação não teve o sucesso que seria de esperar e também porque a implementação não foi totalmente efectuada. Apesar de as acções sugerirem uma abordagem em cascata, tal não aconteceu já que se tiveram que flexibilizar muitas das acções, realizando-as em paralelo.

vi) Monitorização

No:	Performance indicator	Unit	Current base	Future goal	Actual result	Effect
1	Number of Complaints Inserted	Integer	0	All	106	
2						
3						
Key (Effect):			Good: G		Bad: R	
					Okay: Y	

Figura 30. Indicadores de Performance.

Um dos passos do A3 consiste em encontrar indicadores que permitam medir a melhoria, de forma a efectuar uma monitorização dos resultados. Neste caso específico foi difícil encontrar indicadores uma vez que o objectivo é em grande parte qualitativo, pretendendo-se que exista uma gestão das reclamações a 100% com o novo sistema. De forma a medir o uso do sistema teve-se como base o número de reclamações inserido (ver figura 30). No final do período em que decorreu este projecto tinham sido inseridas 106 reclamações, que foram centralizadas e controladas num único local.

vii) Resultados

Até ao fim do período em que decorreu o presente projecto foram controladas, e introduzidas, 106 reclamações no sistema protótipo. No entanto o protótipo funciona de forma parcial, uma vez que só foi testado com um Actor (ver ponto 4.1).

3.2.2. Benchmarking e Estudo de Sistemas Existentes

Como já foi referido no capítulo 2, o *benchmarking* é uma metodologia através da qual se podem identificar novas ideias e melhorar os processos. No presente projecto efectuou-se uma visita à *Bosch Security Systems S.A*, com o intuito de analisar o processo de gestão de reclamações lá existente e implementar as boas práticas na *Bosch Termotecnologia S.A*. Outras visitas estavam planeadas, nomeadamente às organizações C.A.C.I.A, *Blaupunkt* e Autoeuropa, no entanto devido, essencialmente, à conjuntura económica foram impossíveis de realizar. Por outro lado, efectuou-se um estudo a dois sistemas de gestão de reclamações existentes no mercado que permitiu identificar novas ideias para melhoria do processo e construção do sistema.

3.2.2.1. Benchmarking – Bosch Security Systems S.A

O estudo de *benchmarking* efectuado estrutura-se de acordo com a abordagem de Oakland & Sohal (2001).

i) Planeamento

Inicialmente começou-se por seleccionar o processo em estudo (Reclamações aos Fornecedores). A escolha da empresa foi efectuada pela *Bosch Termotecnologia S.A*, que considerou que a empresa tinha interesse neste âmbito. De seguida identificaram-se os principais *benchmarks*, que residem no conhecimento do fluxo de informação, levantamento do processo existente, identificação de possíveis SI's de apoio à gestão de reclamações pretendendo-se, também, saber como possuem o histórico de reclamações e efectuam o controlo dos 8D's. O método de recolha de dados baseou-se em análise qualitativa com comparação dos processos. A recolha de dados baseou-se no acompanhamento do processo existente.

ii) Análise

Após a visita efectuada constatou-se que o *benchmarking* tinha sido mal planeado uma vez que a organização em estudo possuía um sistema bastante menos complexo do que o da *Bosch Termotecnologia S.A*. Este facto pode ser explicado pela diferença na quantidade de componentes, que estão envolvidos, e na própria individualidade organizacional.

Em termos de SI's a *Bosch Security Systems S.A* utiliza uma folha de Excel (ver figura 31) para introduzir as suas reclamações (método que, pela quantidade reduzida de componentes inseridos, não se revela ineficiente), ficando disponível um histórico das quantidades de problemas for componente e fornecedor.

	supplier 1	supplier 2	supplier 3	supplier 4	supplier 5	supplier 6	...
Componente 1							
Componente 2							
Componente 3			Quantidades				
Componente 4							
Componente 5							
Componente 6							
Componente 7							
Componente 8							
Componente 9							
Componente 10							
Componente 11							
Componente 12							
Componente 13							
Componente 14							
Componente 15							
Componente 16							
Componente 17							
Componente 18							
...							

Figura 31. Informatização das Reclamações – *Bosch Security Systems S.A.*

Por outro lado, ao nível dos 8D's existe uma grande preocupação com a acção de contenção, existindo um acompanhamento sistemático até ao passo D3.

Concluindo, nos três pontos principais de comparação verificou-se uma ausência daquelas práticas (ver tabela 6):

Tabela 5. Balanço *Benchmarking* - *Bosch Security Systems S.A.*

Parâmetros Analisados	<i>Bosch Security Systems</i>
Histórico Reclamações	Excel (quantidades e referências).
Controlo dos 8D's	Manual.
Informatização Reclamações	Excel.

Através do acompanhamento de reclamações na empresa foi possível verificar, de uma forma simplificada, que o processo de reclamação (ver figura 32) (embora semelhante ao da *Bosch Termotecnologia S.A* estruturalmente) revela ser mais informal e com menos trocas de dados, sendo essas trocas bastante verbais e sem registo associado.

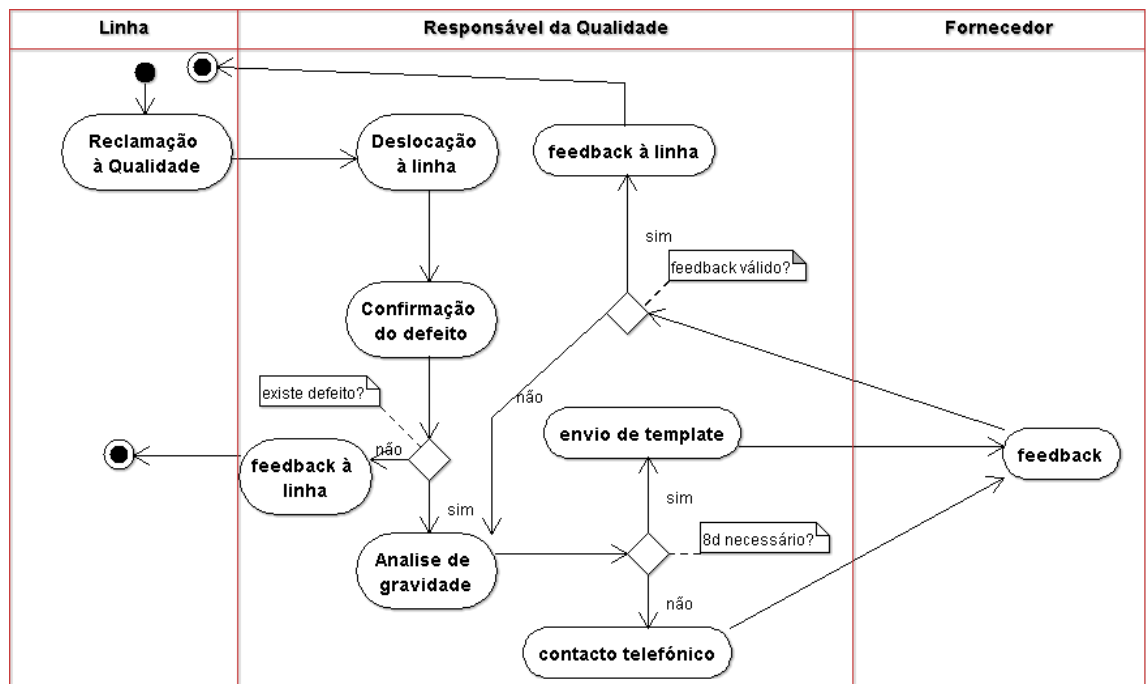


Figura 32. Diagrama de Actividades – *Bosch Security Systems S.A.*

Através deste estudo conclui-se que as organizações possuem diferenças no que toca aos seus processos e métodos de trabalho.

As áreas de negócio das organizações também explicam esta diferença, uma vez que para peças de compra que integram um esquentador tem que existir um cuidado extremo, já que se trata de produtos que funcionam a gás, sendo necessário garantir níveis de segurança elevada. Neste sentido o processo de reclamação é encarado com elevada importância.

3.2.2.2. Estudo de Sistemas Existentes – SupplyOn Vs. IQUIS

O estudo de sistemas existentes permitiu identificar ideias para este projecto através da análise de soluções padronizadas. O estudo acaba por ser baseado em *benchmarking*, uma vez que passa pela análise de soluções conhecidas e comercializadas actualmente, sendo referências na sua área. Neste caso analisaram-se dois sistemas existentes que são o SupplyOn e o IQUIS recorrendo a informação existente na *Bosch Termotecnologia S.A.*

i) Sistema SupplyOn

A aplicação SupplyOn criou-se através da ideia de uma plataforma única para lidar com todos os processos inerentes aos clientes e fornecedores, numa organização, no que concerne aos fluxos de comunicação e informação externos.

O sistema permite efectuar uma gestão de todos os processos organizacionais de forma integrada desde a concepção até à produção (figura 33).

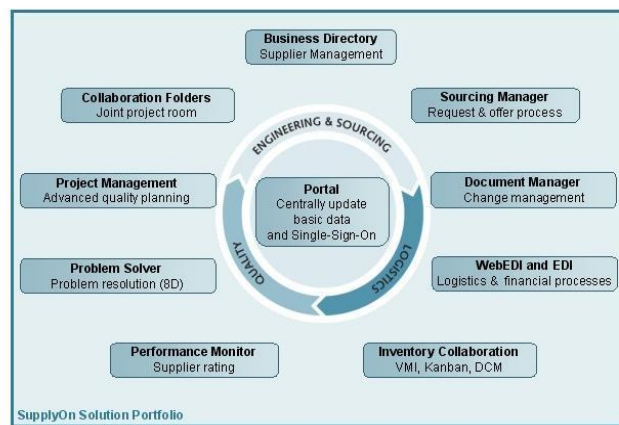


Figura 33. Cadeia do SupplyOn (Bosch (2009)).

A Interface do SupplyOn pode ser baseada num site, bastando ao fornecedor ter acesso à Internet (através de webEDI³) para efectuar a sua gestão de reclamações.

A aplicação permite uma gestão de informação a todos os níveis da organização nomeadamente Compras, Logística, Finanças, Qualidade e Engenharia sendo, nesse aspecto, semelhante ao SAP e outros ERP's existentes no mercado, mas mais vocacionada para troca de informação externa.

A sua vantagem mais notória reside na troca de informação e comunicação com fornecedores e clientes através de uma plataforma comum, permitindo uma maior transparência e conteúdo dos fluxos associados.

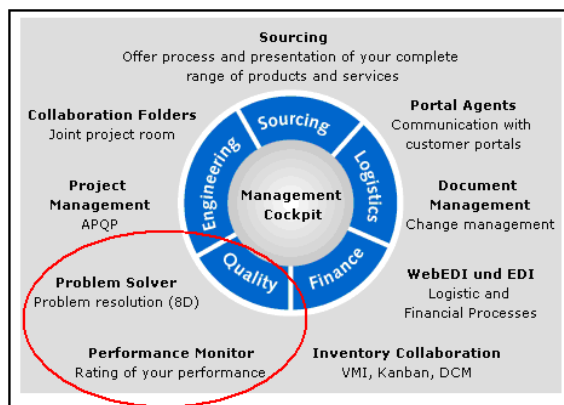


Figura 34. Utilização do SupplyOn nas diferentes áreas funcionais (Bosch (2009)).

Apesar de o SupplyOn apresentar aplicações para todas as áreas funcionais da organização, no estudo em causa apenas interessa as vocacionadas para a Qualidade (ver figura 34).

³ WebEDI- Troca estruturada de dados via Internet.

Nesta área as maiores valências da aplicação são ao nível da resolução de problemas (8D) e da monitorização da Performance que os fornecedores podem ter de si próprios, sabendo em tempo real os seus PPM's⁴ e outros dados.

Uma das ferramentas desta aplicação que mais pode auxiliar o PUQ reside na gestão de reclamações. O sistema SupplyOn permite tratar as reclamações de forma rápida e assegurar que os Fornecedores respondem às mesmas. O processo funciona da seguinte forma:

- i) As reclamações são transferidas do sistema de gestão de qualidade interno, via SupplyOn, para os fornecedores.
- ii) Os fornecedores respondem à reclamação no formato 8D e o sistema fornece informação sobre o *status* da reclamação.
- iii) Existe um sistema de *reminders*, tanto para o PUQ como para os fornecedores, que vai sinalizando as reclamações até que o problema seja resolvido.

Tem a vantagem de ser um sistema consultado bilateralmente – PUQ/Fornecedor - e de ser dinâmico, isto é, aberto à troca de informação entre diferentes organizações. Enquanto sistemas internos só permitem uma actualização e o manuseamento de informação dentro da própria organização, no caso do SupplyOn temos uma interacção com o exterior que pode poupar tempo e permitir que os fluxos de informação sejam mais coerentes. Esta gestão de reclamações tem como vantagens: os fornecedores como o PUQ terem acesso à mesma informação a toda a hora; evitar-se redundância de informação; reduzirem-se os custos de transacção; e a transacção de Informação entre PUQ/Fornecedor ser mais eficiente.

ii) Sistema IQUIS

Por outro lado, o IQUIS é um sistema de qualidade integrado constituído maioritariamente por componentes do SAP. O âmbito o IQUIS é o seguinte:

- ✓ Interdepartamental.
- ✓ Reclamações externas e internas
- ✓ Reclamações técnicas e logísticas.
- ✓ Reclamações de linha.

⁴ PPM – Part per million – neste caso refere-se a problemas de qualidade.

Foi feito para efectuar um registo eficiente de reclamações externas e internas podendo-se ao mesmo tempo controlar os 8D's respectivos. A vantagem desta ferramenta é permitir, de um modo integrado, a oportunidade de gestão de todas as reclamações afectas à cadeia de valor.

Neste caso não existe comunicação com o fornecedor, sendo uma ferramenta apenas de cariz interno, ao contrário do SupplyOn que apresenta uma opção mais dinâmica. Como tal, este sistema não apresenta grande valor para este projecto. Por outro lado, e uma vez que se trata de uma ferramenta baseada no ERP SAP, pode traduzir-se em dificuldade de implementação.

iii) SupplyOn Vs. IQUIS

Na tabela seguinte apresenta-se um comparativo entre os dois sistemas no que toca às suas vantagens e desvantagens (ver tabela 6):

Tabela 6. SupplyOn Vs. IQUIS – Vantagens e Desvantagens.

	SupplyOn	IQUIS
Vantagens	<ul style="list-style-type: none"> Interface intuitivo – baseado Internet. Interacção Fornecedor – PUQ. Alertas dos 8D's e reclamações pendentes ao Fornecedor. Tecnologia acessível em todo o mundo. 	<ul style="list-style-type: none"> Integrado com o SAP. Ambiente SAP familiar a muitos utilizadores. Possível gerir reclamações.
Desvantagens	<ul style="list-style-type: none"> Inexistência de histórico – tem que estar ligado a sistema interno da empresa. Não caminha no sentido da integração com o SAP. Necessidade de tecnologia EDI ou webEDI obrigatoriamente para todas as partes – custos elevados. 	<ul style="list-style-type: none"> Pouco dinâmico. Vocacionado para o sector automóvel. Ferramenta potencialmente complicada de utilizar – Formação em SAP. Custos com ferramentas SAP elevados.

Através deste estudo, realizado ao longo do processo de análise e levantamento de requisitos, surge, então, a ideia da solução proposta, bem como os dados que irão definir o modelo conceptual daquela solução.

3.3. Solução Proposta

Após levantamento dos requisitos e do processo de negócio existente, passou-se à modelização da solução que veio a compor o sistema. Antes da apresentação da solução propriamente dita, ir-se-á fazer referência ao processo que se espera vir a obter com a introdução da nova solução.

Pois, tal não implica (ou significa) uma reengenharia de processos, mas sim uma optimização do processo existente, já que uma das grandes valências dos SI's, reside na automatização de tarefas e na imposição de *standards* que permitem tornar os processos transparentes e com o fluxo de informação optimizado.

De seguida descreve-se então o processo optimizado e que se espera vir a encontrar com a nova solução. O mesmo pode ser vista na sua forma gráfica através no diagrama de actividades representado na figura 35:

1. Surge um problema que gera uma reclamação, feita pela Origem, para a Qualidade das Compras.
2. A reclamação é analisada pelo responsável, do componente em causa, que pode ou não validar a reclamação, gerando a situação 'a ou b.
 - a. Se a reclamação for válida é analisada a gravidade do problema para avaliar se existe necessidade de pedir ou não um 8D, gerando a situação a.1. ou a.2.
 - a.1. Se for necessário pedir um 8D, o mesmo é validado, passando-se para o passo a.1.1.
 - a.1.1. O fornecedor responde ao 8D, passando-se para o ponto a.1.1.1.
 - a.1.1.1. O responsável valida o 8D podendo ocorrer duas situações, a.1.1.1.1 ou a.1.1.1.2.
 - a.1.1.1.1. Se o 8D não for válido, volta para o ponto a.1.
 - a.1.1.1.2. Se o 8D for válido, vai para o ponto 3.
 - a.1.1.2. Se o 8D for válido, vai para o ponto 3.
 - a.2. Se não for necessário pedir um 8D, o responsável pede acções normalmente, validando as mesmas.
 - a.2.1. O fornecedor responde às acções pedidas passando para o ponto a.2.2.1
 - a.2.2.1. O responsável valida as acções podendo ocorrer duas situações, a.2.2.1.1 ou a.2.2.1.2.
 - a.2.2.1.1. Se as acções não forem válidas, volta para o ponto a.2.
 - a.2.2.1.2. Se as acções forem válidas, vai para o ponto 3.
 - b. Se a reclamação não for válida termina o processo.
 3. A origem aprova os resultados podendo ocorrer duas situações, 3.a. ou 3.b.
 - 3.a. Se existir ausência de defeito a origem fecha a reclamação.
 - 3.b. Se existir defeito volta-se ao ponto 1.

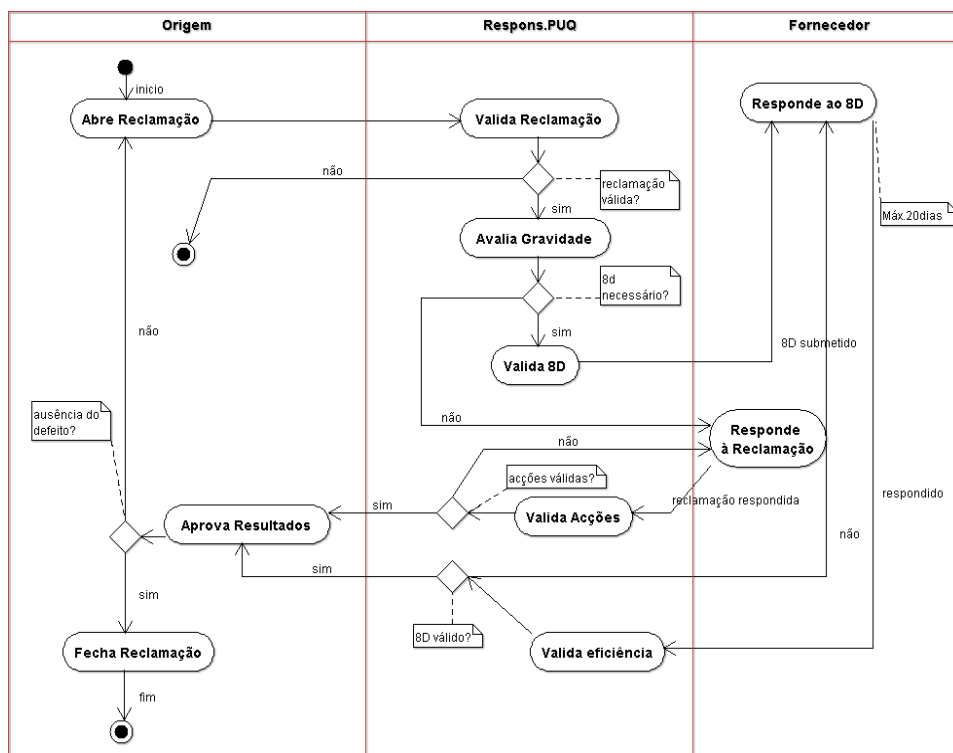


Figura 35. Diagrama de Atividades - Após Sistema.

Como se pode verificar no diagrama da figura 35 algumas tarefas são automatizadas, sendo desnecessário enviar templates ao fornecedor e contactá-lo por telefone ou correio electrónico a requerer acções, uma vez que a actual proposta considera que essas tarefas são efectuadas através do sistema.

Ao mesmo tempo propõe-se a existência de um único canal de circulação da informação que evita a dispersão de informação por conversas telefónicas, emails ou templates dispersos, o que proporcionará uma garantia de histórico dos problemas, centralização dos mesmos num único local e uma melhoria na qualidade da comunicação.

3.3.1. Modelo Conceptual

De acordo com Batini et al. o desenho de base de dados inicia-se com a solução conceptual. Esta começa com a especificação dos requisitos e resultados num esquema conceptual da BD, em que o esquema conceptual representa uma descrição de alto nível da estrutura da BD com o propósito descrever o tipo de informação da mesma. Neste sentido, e de encontro ao que o autor propõe, começa-se por definir o Modelo conceptual, fazendo uso dos diagramas de casos de utilização, de classes e de sequência.

3.3.1.1. Diagrama de Casos de utilização

Com os processos definidos, e o conhecimento dos requisitos, chegou-se ao diagrama de casos de utilização geral do sistema. Através deste é possível representar todas as funcionalidades disponíveis aos utilizadores do sistema. Este diagrama permite definir as fronteiras do sistema e especificar os requisitos de acordo com a perspectiva dos diferentes perfis de utilizadores.

Naiburg & Maksimchuk (2001) referem que a suportar o modelo de casos de utilização devem existir descrições que retratem a informação básica e fluxos relativos ao caso. Neste sentido apresenta-se uma breve descrição dos principais actores envolvidos, bem como uma descrição dos casos de utilização e propósito dos mesmos (ver tabela 7).

- ✓ **Origem** – representa todas pessoas que efectuam reclamações ao departamento PUQ.
- ✓ **Respons.PUQ** – Representa o responsável de Engenharia do PUQ que trata uma determinada reclamação.
- ✓ **Fornecedor** – Representa o Fornecedor responsável pela venda do componente que apresenta problemas.

Tabela 7. Lista de Casos de utilização.

Nome do caso de utilização	Actor responsável	Propósito do caso de utilização
Abrir Reclamação	Origem	Abertura de uma nova reclamação no sistema. Este actor deve entrar no sistema e inserir os dados relativos ao problema que encontrou no seu sector.
Validar Reclamação	Respons.PUQ	O responsável pelo componente em causa deve analisar a reclamação e efectuar a sua validação de forma a mesma entrar no sistema.
Pedir 8D	Respons.PUQ	Caso se trate de um problema crítico e responsável deve efectuar o pedido de um 8 D ao fornecedor após a validação.
Responder Reclamação	Fornecedor	O fornecedor deve responder às suas reclamações pendentes no sistema introduzindo as acções efectuadas.
Responder 8D	Fornecedor	O fornecedor deve responder aos 8D's existentes no sistema.
Validar Acções	Respons.PUQ	O responsável PUQ deve validar as acções do fornecedor para que sejam aceites no sistema.
Validar Eficiência	Respons.PUQ	O responsável PUQ deve validar a eficiência do 8D para que este fique aceite.
Aprovar Resultados	Origem	A origem verifica os resultados e pode reabrir a reclamação, se entender que o problema continua a existir.
Fechar Reclamação	Origem	A origem verifica os resultados e pode fechar a reclamação se entender que o problema está finalmente resolvido.

Após a definição dos actores e dos casos de utilização, chegou-se ao seguinte diagrama (ver figura 36):

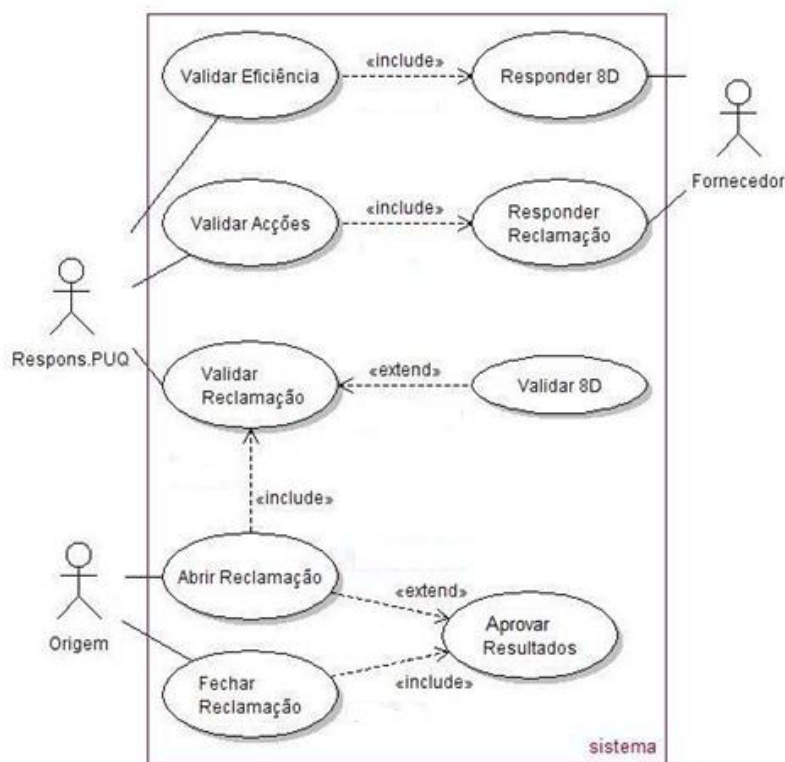


Figura 36. Diagrama de Casos de utilização para o sistema.

Este tipo de diagrama permite traduzir os aspectos dinâmicos do sistema identificando o papel de cada actor no mesmo.

É possível encontrar relações de dependência do tipo *extend* e *include*. No caso exemplificado a seguir (ver figura 37) temos uma relação *extend* porque nem sempre quando se valida a reclamação se tem que validar um 8D. O Responsável do PUQ apenas valida o 8D se achar necessário, consoante a gravidade do problema.

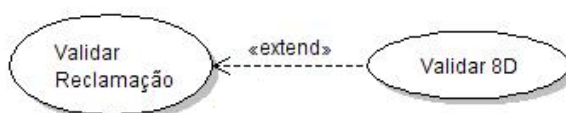


Figura 37. Exemplo de relação *extend*.

Na figura 38, temos o exemplo de uma relação *includes*, que implica obrigatoriedade. Neste caso para o Responsável PUQ poder validar as acções, o Fornecedor tem que ter respondido às mesmas.

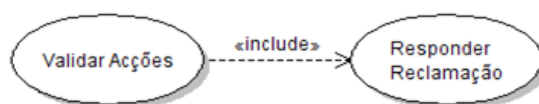


Figura 38. Exemplo de relação *include*.

Outros diagramas de casos de utilização poderiam ter sido efectuados, no entanto apenas se pretende demonstrar de uma forma as principais funcionalidades do sistema sem detalhar minuciosamente todas as acções existentes.

3.3.1.2. Diagrama de Classes

O passo seguinte consistiu na construção do diagrama de classes de forma a obter a estrutura do SI, segundo os requisitos estabelecidos. Um conjunto de elementos, nomeadamente classes, atributos e relações foram encontrados no domínio do problema cuja especificação se descreve da seguinte maneira:

“Uma determinada origem pode efectuar múltiplas reclamações, ao longo do tempo, mas uma reclamação apenas pode advir de uma origem.

Da mesma forma um responsável trata de muitos componentes (classe material) mas para cada componente corresponde um só responsável logo, de cada vez uma reclamação só pode ser analisada e validada por um responsável. Um responsável apenas possui um substituto mas um substituto pode dar resposta a vários responsáveis.

Uma reclamação apenas pode gerar 0 ou 1 8D, e um 8D corresponde sempre a uma reclamação. Para uma reclamação pode ser necessário inserir várias imagens logo uma reclamação pode conter muitas imagens, mas uma imagem corresponde unicamente a uma reclamação. Uma reclamação pode apenas possuir um *status* mas um *status* está para muitas reclamações.

Um fornecedor é responsável por vários componentes (referências de material), no entanto um componente existe apenas num fornecedor. Uma reclamação apenas corresponde a um componente (referência de material) mas um componente pode constar em múltiplas reclamações. Então, para uma reclamação só pode corresponder um fornecedor e um componente. Uma classe pode estar para vários fornecedores, mas um fornecedor apenas pode ter uma classe.

Uma reclamação pode conter vários anexos mas um anexo apenas corresponde a uma reclamação. Uma reclamação está para um grau de importância, que define a sua gravidade, mas um grau de importância está para muitas reclamações.

Uma reclamação pode estar para várias acções do fornecedor (pode existir acção de contenção, correctiva, preventiva, etc.).

Mas cada acção pode corresponder apenas a uma reclamação e consequentemente a um tipo de material, fornecedor e responsável. Um tipo de acção pode estar para várias acções do fornecedor mas uma acção do fornecedor apenas está para um tipo de acção.

Da mesma forma, um 8D pode corresponder a várias acções do fornecedor mas uma acção do fornecedor corresponde a apenas um 8D e uma acção do fornecedor corresponde apenas a um tipo de acção. Por outro lado um tipo de acção de 8D corresponde a muitas acções.

Um código de falha pode estar para várias referências de material, mas uma referência de material apenas possui um código de falha. Um subgrupo (*failure classification*) de códigos de falha está para muitos códigos de falha mas um código de falha apenas está para um subgrupo.”

Com base nesta descrição encontrou-se o modelo conceptual de dados concretizado através do diagrama de classes da notação UML (ver figura 39). O respectivo dicionário de dados pode ser consultado no anexo 4.

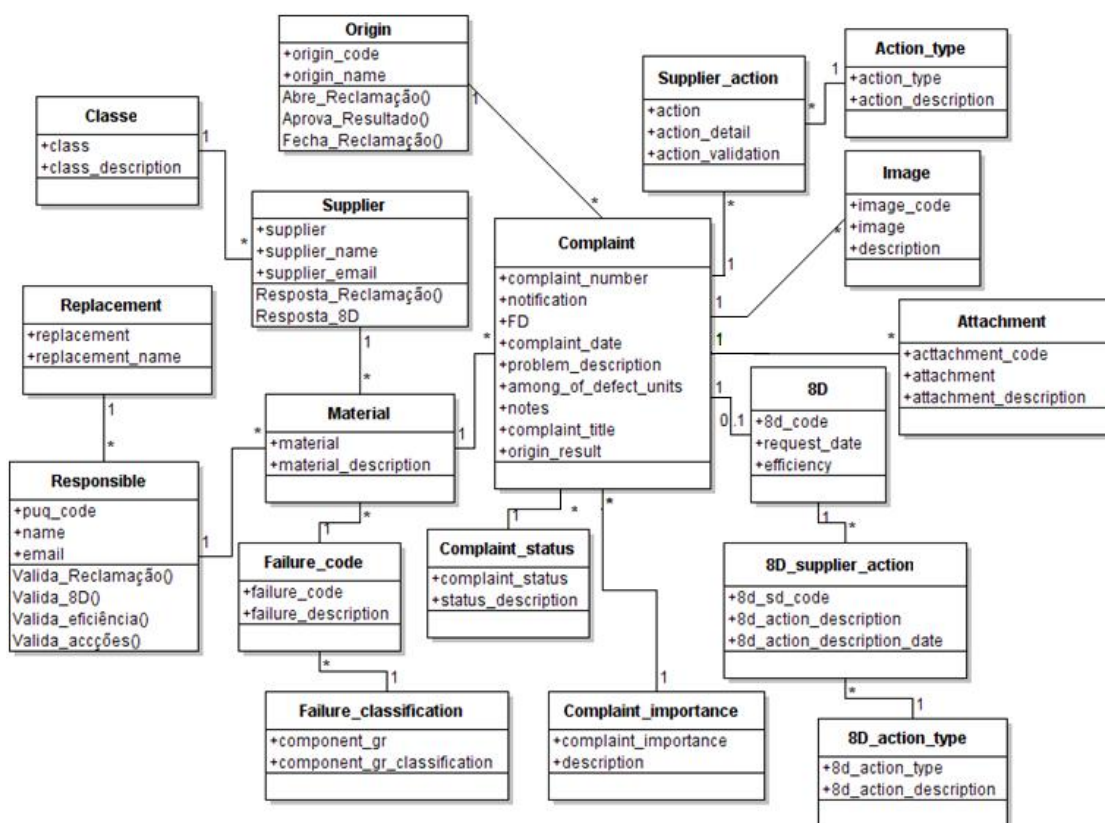


Figura 39. Diagrama de Classes do Modelo (Forma ampliada em anexo 5).

3.3.1.3. Diagrama de Sequência

Para compreender melhor a dinâmica, do sistema, no tempo podem-se efectuar diversos diagramas de sequência. Nestes diagramas o tempo decorre de cima para baixo sendo o diagrama constituído por objectos, linhas de vida do objecto e mensagens.

Estes diagramas revelam as interacções entre objectos num determinado período de tempo.

A figura 40 exemplifica um diagrama de sequências que considera uma reclamação efectuada pelo actor Origem, partindo do pressuposto que a 'Origem' existe no sistema.

Analisando a figura pode-se verificar que, para uma origem efectuar uma reclamação, acede ao interface, e este por sua vez consulta a classe 'origin' para verificar a sua existência, e só posteriormente existe a criação da reclamação, utilizando para isso a classe 'complaint', onde se introduzem os dados relativos à mesma. Após estes acontecimentos a classe 'complaint' retorna a mensagem, de conclusão da reclamação, à classe 'origin' que por sua vez retorna à Interface e este último à 'Origem'.

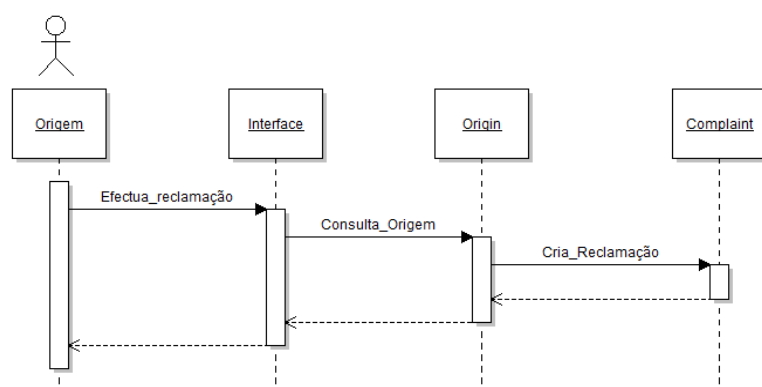


Figura 40. Diagrama de sequência – Actor Origem a efectuar Reclamação.

Desta forma é mais fácil compreender a evolução dos acontecimentos no tempo.

Uma vez que o modelo conceptual de dados assentou num paradigma orientado a objectos e a implementação baseou-se no modelo relacional, houve a necessidade de converter o modelo conceptual encontrado e representado na figura anterior, para modelo relacional. Para tal contaram-se com um conjunto de regras de transposição já existentes para esse fim.

3.3.2. Modelo Lógico

De acordo com Batini et al. (1992) o desenho lógico inicia-se a partir do esquema conceptual e resulta num Modelo lógico, sendo este uma representação da estrutura da BD que pode ser processada num SGBD.

Seguidamente apresenta-se o modelo lógico que derivou da transposição do diagrama de classes encontrado neste projecto (ver figura 39).

3.3.2.1. Modelo Relacional

Com o diagrama de classes construído é fácil obter o modelo lógico, mais especificamente modelo relacional, utilizando para tal as regras de transposição existentes.

Como no diagrama de classes apenas se têm associações do tipo “um para muitos”, e “um para um”, não foi necessário utilizar a totalidade das regras existentes, mas apenas uma parte delas. De seguida apresentam-se as regras a utilizar de acordo com Costa (2004):

- ✓ As tabelas resultam de classes do modelo de classes e associações de “muitos para muitos”.
- ✓ Todas as tabelas têm que ter uma chave primária (ver tabela 8). A notação utilizada considerou-se PK = *Primary Key* e FK = *Foreign Key*.

Tabela 8. Lista de tabelas e chaves primárias.

Tabela	Chave Primária
Classe	class
Replacement	replacement
Responsible	puq_code
Failure_code	failure_code
Failure_classification	component_gr
Origin	origin_code
Supplier	supplier
Material	material
Complaint_status	complaint_status
Complaint	complaint_number
Complaint_importance	complaint_importance
Supplier_action	action
8D	8d_code
8D_supplier_action	8d_sd_code
Action_type	action_type
Image	image_code
Attachment	attachment_code
8D_action_type	8D_action_type

- ✓ Nas associações de um para um a classe que tem a menor número de ocorrências herda a chave estrangeira (ver figura 41).

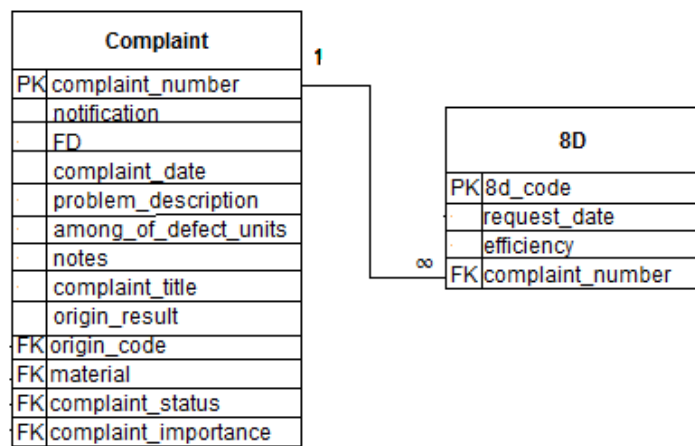


Figura 41. Exemplo de transposição – “um para um”.

Neste caso a tabela 8D pode ter 0 ou 1 ocorrência para cada reclamação (*complaint*) por isso herda a chave estrangeira complaint_number.

- ✓ Nas associações de “um-para-muitos”, as tabelas cujos registros sejam endereçados diversas vezes herdam a referência cuja correspondência é unitária, isto é, a chave estrangeira irá para a tabela “muitos” (ver figura 42).

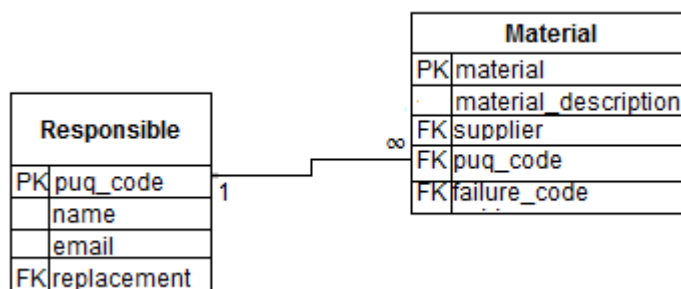


Figura 42. Exemplo de transposição – “um para muitos”.

Nesta situação, como um responsável se pode ocupar de múltiplas reclamações a chave primária irá para a tabela material sob a forma de chave estrangeira.

Após transposição da totalidade do modelo conceptual, segundo as regras existentes, obteve-se o seguinte modelo relacional (ver figura 43).

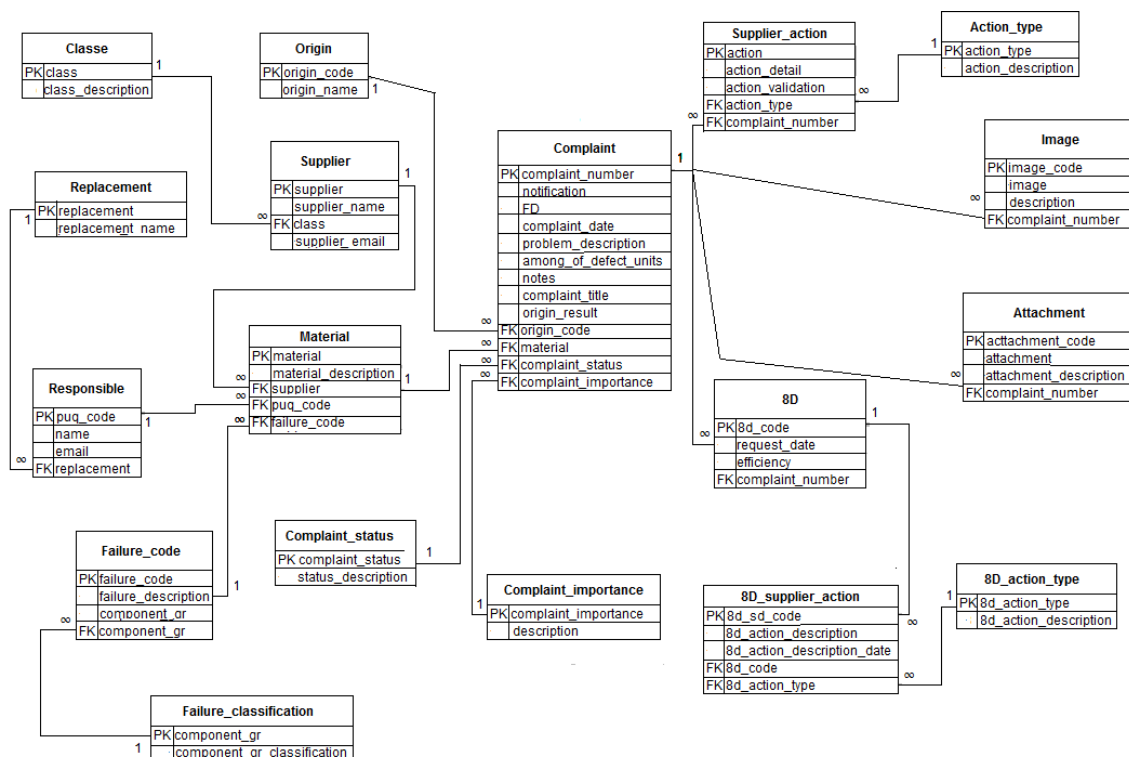


Figura 43. Modelo relacional – Sistema de Gestão de Reclamações (Forma ampliada em anexo 6).

Após a obtenção do modelo relacional, o passo seguinte do projecto foi dado no sentido de implementar aquele modelo num SGBD à escolha, vindo-se assim a construir o protótipo num SGBD em *MS-Access*, que também serviu para testar o modelo relacional em ambiente real.

3.3.3. Modelo Físico

Batini et al. (1992) refere que o desenho físico é a última etapa do processo de desenho de uma BD. Refere que, o mesmo, é efectuado através do esquema lógico e resulta no esquema que descreve as estruturas de armazenamento e métodos de acesso utilizados para aceder aos dados. Neste sentido através do esquema lógico (modelo relacional) definiram-se as estruturas de armazenamento utilizando o software *MS-Access*, finalizando-se o desenho da BD.

3.3.3.1 O Protótipo

Após a modelização efectuada é necessário testar os modelos desenvolvidos. Uma vez que o presente projecto se desenrolou durante um espaço de tempo limitado teve que se efectuar o protótipo num sistema acessível e que permitisse uma construção rápida.

Chegou-se à conclusão que face aos recursos disponíveis a opção seria o *MS-Access*. De acordo com Carriço (1996) a arquitectura do SGBD *MS - Access* é a seguinte:

- i) A BD propriamente dita constituída por um conjunto de objectos tais como tabelas, consultas, formulários, relatórios e modules sendo a BD armazenada num ficheiro com extensão “mdb”.
- ii) A *engine* da BD, constituída por um conjunto de programas responsáveis pela execução de todas as operações de acesso a BD. Actua como gestor de dados (*data manager*).
- iii) O interface gráfico, que disponibiliza o acesso dos utilizadores a criação de tabelas, definição de consultas, criação de relatórios e formulários de apresentação de dados.

A construção do Sistema em *MS-Access* efectuou-se de acordo com os pontos referidos por Carriço (1996), mais concretamente:

- i) Construção das tabelas do sistema (de acordo com o modelo).
- ii) Estabelecimento de associações entre as tabelas (de acordo com o modelo).
- iii) Construção das consultas necessárias.
- iv) Construção dos relatórios necessários.
- v) Construção dos Formulários e Programação de Acções / Restrições – Desenho de Interface tendo como orientação a metodologia *L.U.C.I.D.*
- vi) Construção do manual de ajuda.

Um exemplo com recurso a um interface do protótipo pode ser visto na figura seguinte (ver figura 44), em que o actor Origem pode inserir a reclamação e o actor Respons.PUQ pode validar ou não a mesma.


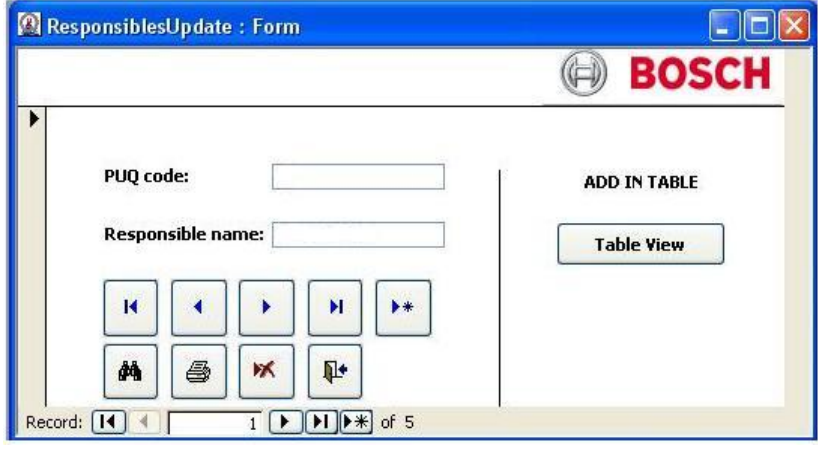
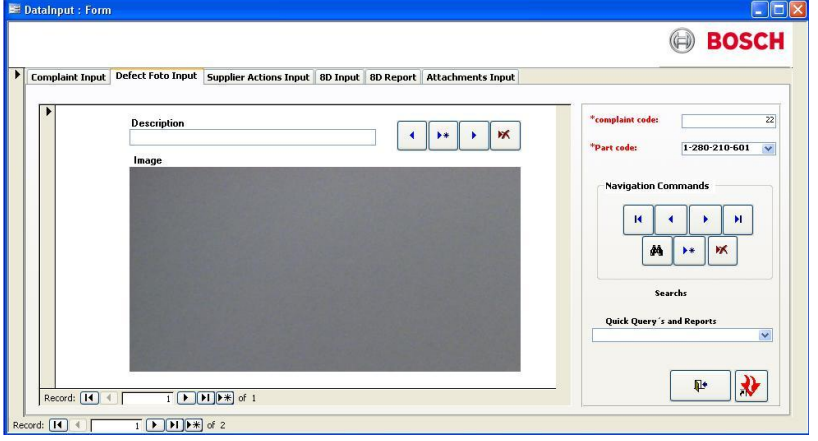
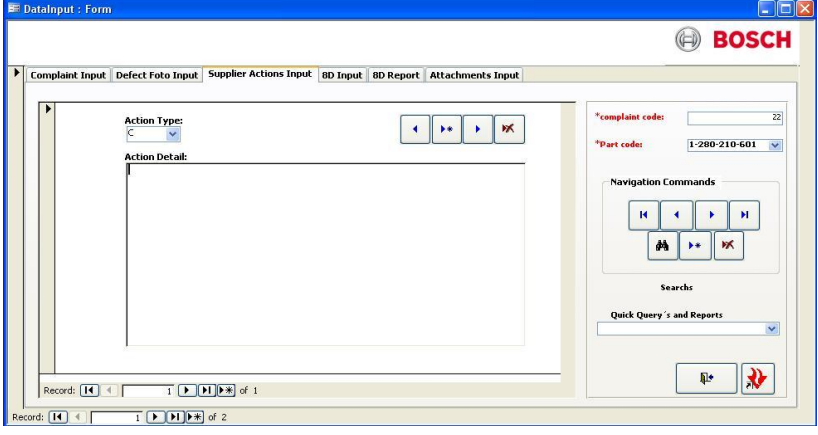
Figura 44. Menu de Inserção da Reclamação – Origem.

Este interface baseou-se em outros sistemas *Bosch*, o que garantiu familiaridade de notações e aspecto, tendo-se uma construção iterativa que implicou bastantes alterações no mesmo consoante os testes que os utilizadores iam efectuando.

De seguida encontram-se outros exemplos de *print screens* da interface que resultou do protótipo (ver tabela 9):

Tabela 9. Exemplos de print screens da Interface.

Interface	Propósito
	<p>Menu que surge inicialmente ao iniciar o sistema. Permite aos diferentes actores entrarem no sistema para efectuarem a gestão das reclamações e ao actor Respons.PUQ efectuar a actualização de dados raiz.</p>


	<p>Interface que permite ao actor Respons.PUQ aceder às tabelas para actualização de dados (associado a código visual Basic com evento clique).</p>
	<p>Menu que permite ao actor Respons.PUQ efectuar a actualização da tabela 'Responsibles'. Surge após clicar na opção 'Responsibles' do menu anterior.</p>
	<p>Menu que permite, ao actor Origin, a inserção de múltiplas fotos associada a uma reclamação.</p>
	<p>Menu que permite, ao actor Supplier, a inserção das suas acções.</p>

Este protótipo tem como objectivo validar o modelo relacional e verificar se o mesmo cumpre os requisitos para os quais foi concebido. No entanto é de referir que o modelo na totalidade teria que se envolver tecnologia Web que permitisse o acesso os fornecedores.

Algumas acções foram programadas de forma a evitar um uso incorrecto da BD (ver tabela 10), tendo utilizado nesta fase o *Visual Basic* (ver anexo 7), uma programação orientada a eventos. Seguidamente apresentam-se exemplo de uma acção associada a um botão:

Um exemplo destes existe no caso em que o utilizador tenta abrir um novo registo sem que tenha preenchido adequadamente o registo actual (ver tabela 10). Neste caso o sistema impõe o preenchimento de campos obrigatórios, aparecendo a mensagem *"Please fullfill all the red fields!"* caso não sejam preenchidos os campos a vermelho.

Tabela 10. Exemplo de programação associada a botão.

Interface	Acção	Evento	Código Visual Basic
	Clique	Surge mensagem caso os campos obrigatórios não tenham sido preenchidos.	<pre> Private Sub Command76_Click() On Error GoTo Err_Command76_Click If Me!complaint_title <> 0 And Me!problem_description <> 0 And Me!material <> 0 Then DoCmd.Close Exit_Command76_Click: Exit Sub Err_Command76_Click: MsgBox Err.Description Resume Exit_Command76_Click Else MsgBox "Please fullfill all the red fields!" Exit Sub End If </pre>

Neste projecto começaram-se por efectuar as consultas obtendo-se através destas os relatórios necessários (ver anexo 8). Desta forma conseguiram-se gerar listagens com informação útil para os Actores.

Ao construir o manual teve-se o cuidado de efectuar uma abordagem por processos, tentando-se detalhar os mesmos ao máximo para que o utilizador os compreenda.

Um dos pontos da norma 9001:2008 refere-se à abordagem por processos o que se compagina com o que é feito durante todo este projecto. Um dos processos no manual encontra-se em Anexo (ver anexo 9).

Uma das vantagens do Access é o facto de ser uma ferramenta de simples utilização, permitindo testar vários cenários em pouco tempo sem que sejam necessários conhecimentos profundos de programação, o que coloca esta ferramenta numa posição de vantagem em relação ao desenvolvimento de protótipos.

No entanto o Access se por um lado é de fácil utilização, por outro também apresenta algumas desvantagens face a outros SGBD mais robustos, como por exemplo no que toca à sua capacidade de armazenamento (cerca de 2 *gigabytes* já com a estrutura da base de dados incluída), e na ineficácia do funcionamento com utilizadores simultâneos.

3.3.4. SQL

Dadas as limitações do Access, foram criadas as instruções SQL (ver anexo 10) que permitem implementar o mesmo modelo num outro SGBD, como por exemplo o *Oracle*, que colmate os pontos negativos do Access. Através da DDL (*Data Definition Language*) escreveram-se as instruções para construção das tabelas, sendo de seguida apresentado um exemplo referente à criação da tabela 'Complaints'.

CREATE TABLE Complaints

```
(
    complaint_number int(10) not null auto_increment primary key,
    notification int(10) not null,
    FD int(4) not null,
    complaint_date datetime not null,
    problem_description varchar(250) not null,
    among_of_defect_units int(10) not null,
    notes varchar (100) not null,
    complaint_title varchar (50) not null,
    origin_result
    material int(10) not null references Material (material),
    origin varchar(1) not null references Origins (origin_code),
    complaint_importance varchar(1) not null references Complaints_importance
    (complaint_importance),
);
```

As outras tabelas são feitas de forma semelhante.

Através da DML (*Data Manipulation Language*) pode-se interrogar a BD construída de forma a obter as consultas necessárias. Por exemplo, para obter a listagem de Reclamações em aberto podemos efectuar a seguinte instrução:

```
SELECT complaint_number  
FROM Complaints  
WHERE current_status = 'Open'
```

Por outro lado se fosse necessário listar as reclamações referentes a um determinado tipo de material (123456) entre o início de 2008 e 2009 teríamos:

```
SELECT Complaints.complaint_number, Material.material  
FROM Complaints, Material  
WHERE material = '123456%'  
        complaint_date > 2008-01-01 AND  
        complaint_date < 2009-01-01
```

Como já foi referido podem-se também eliminar e adicionar registos utilizando a DML. Se pretendêssemos inserir um registo na tabela Origins poderíamos fazê-lo da seguinte forma:

```
INSERT INTO Origins  
(origin_code, origin_name)  
VALUE ('L', 'Linha')
```

Por outro lado, se fosse necessário eliminar um determinado componente da tabela Material teríamos:

```
DELETE FROM Material  
WHERE material = "123456"
```

Através de comandos simples consegue-se então implementar o modelo relacional num SGBD e efectuar todas as interrogações ou actualizações necessárias.

4. Conclusões, Limitações e Perspectivas de Trabalho Futuro

Este trabalho permitiu desenvolver com sucesso um protótipo de um sistema de Gestão de Reclamações. Embora tenha sido um projecto bastante ambicioso, foi dado um avanço significativo nesta área, conseguindo-se abordar o problema existente de forma estruturada, nomeadamente com recursos a metodologias como são o *benchmarking*, a UML, a abordagem por processos e a ferramenta A3.

O principal objectivo do trabalho consistiu em analisar um problema existente e fornecer uma acção imediata para o mesmo, tarefa que foi conseguida através da construção do protótipo, que surge como uma nova ferramenta para utilizar neste processo de negócio.

Verificou-se que através da UML é possível abordar o desenvolvimento de um SI, recorrendo para isso apenas a alguns diagramas essenciais para retratar as funcionalidades, estrutura e processos do sistema.

No presente projecto foi possível constatar que as disciplinas não actuam isoladas, podendo-se relacionar o desenvolvimento de SI's e a qualidade no que toca ao ponto comum de abordagem por processos e sua melhoria. Ao mesmo tempo o ciclo PDCA vai de encontro às metodologias de desenvolvimento iterativas nas quais a UML se enquadra. Este projecto é exemplo de uma melhoria de um processo através do desenvolvimento de um SI.

Foi também provado que o uso de ferramentas da qualidade é possível ao desenvolver SI's, tendo-se utilizado a ferramenta A3 para levantamento dos requisitos.

Utilizou-se o *benchmarking* como metodologia para o estudo de melhores práticas de outras organizações, actividade que ficou aquém das expectativas devido ao planeamento efectuado. Por outro lado, estudaram-se ferramentas existentes que permitiram obter ideias para este estudo, nomeadamente ao nível da comunicação entre as diferentes partes.

Verificou-se que a abordagem por processos é uma mais-valia na medida em que permite modelizar os problemas e identificar as oportunidades de melhoria.

Como referido no capítulo 2, qualidade significa ir de encontro aos requisitos do cliente, significado que é extensível aos SI's que visam atingir o mesmo objectivo.

Através das regras de transposição conseguiu-se transpor a modelização em UML para o modelo relacional o que permitiu implementá-lo num SGBD. Não obstante o SGBD utilizado ter algumas limitações abordou-se a SQL como forma de transpor esse

problema, uma vez que de uma forma relativamente fácil se pode implementar o modelo para um SGBD mais robusto.

Apesar de o sistema não ter sido completamente implementado, uma vez que os fornecedores não têm ainda a sua plataforma, conseguiu-se testar o modelo relacional efectuado através da sua construção em Access e também os requisitos pretendidos pela Bosch.

Esta solução permite personalizar a ferramenta à medida das necessidades da organização o que não acontece com as soluções estandardizadas no mercado, analisadas no levantamento de requisitos, que além disto possuem custos elevados de aquisição.

O uso de um SI permitirá à organização desfrutar de um fluxo de informação coerente e de uma base histórica para análise dos fornecedores e problemas.

Até ao término do estágio foram inseridas 106 reclamações no sistema, gestão efectuada pelos responsáveis do PUQ, que puderam aceder à informação de uma forma mais automática e desfrutar de tarefas e consultas automatizadas.

O presente trabalho de investigação cumpriu os objectivos propostos pela organização. O protótipo desenvolvido é exemplo disso, uma vez que permite já aos Eng.ºs de Qualidade efectuarem a gestão das suas reclamações. No entanto, existiram algumas limitações no presente projecto de investigação.

Relativamente aos estudos de *benchmarking*, no futuro estes deveriam ser efectuados baseados em dados quantitativos e com organizações com desempenho superior e não de uma forma qualitativa, como neste projecto. O *benchmarking* foi efectuado apenas numa outra organização. A escolha das organizações foi efectuada com algum factor de incerteza, uma vez que não se sabia se as práticas seriam efectivamente as melhores, o que de acordo com a literatura deve acontecer. Oakland e Sohal (2001) afirmam que na fase de planeamento do *benchmarking* é necessário identificar o melhor competidor, utilizando o *feedback* dos clientes ou observadores industriais.

Ficou a faltar implementar o sistema ao nível dos 3 actores, o que será uma tarefa bastante difícil dado a quantidade de fornecedores existentes e pelo facto de terem que possuir, obrigatoriamente, *Internet*.

O sistema prototipado tem que ser ligado a uma interface *Web* de forma a conectar os fornecedores sendo conveniente, também, implementar o modelo num software mais robusto como *Oracle*, por exemplo, uma vez que o Access apresenta as limitações, já discutidas anteriormente, ao nível de capacidade de armazenamento e baixa

performance com utilizadores simultâneos. Para colmatar esta lacuna pode-se implementar o sistema utilizando instruções SQL, como apresentado no projecto.

Esta tarefa não foi desenvolvida em grande parte devido ao curto tempo do projecto e pelo facto de ser uma área que envolve autorizações da Fábrica Central e aquisições de licenças para novos *softwares*, sendo processos que envolvem custos e autorizações que não foram contempladas para este projecto.

Por outro lado a utilização deste *software* não se compagina com o sentido de integração vivido na organização, uma vez que não se integra com o ERP utilizado na organização, situação que pode ocorrer apenas se for implementado no SAP. Actualmente a *Bosch Termotecnologia S.A* encontra-se numa fase de transição de SI's estando a migrar do seu sistema antigo (Winmenu) para uma solução integrada – ERP SAP, solução que não é compatível com o SI desenvolvido no âmbito deste projecto, uma vez que teria que se implementar o modelo relacional no ERP. No entanto, a única solução para este problema poderá passar pela integração do modelo conceptual no ERP, o que se poderá revelar bastante dispendioso e com as desvantagens similares ao *software* IQUIS.

Por outro lado, importava também durante a implementação total, efectuar avaliações da satisfação dos clientes (origens, respons.PUQ e fornecedores), de forma a melhorar continuamente o sistema para que o mesmo cumprisse na totalidade as expectativas dos clientes.

5. Referências Bibliográficas

Aalst, Van W., Hee, Van K. (2002), "Workflow Management – Models, Methods, and Systems", The MIT Press Cambridge.

Apa (2008), "Diagrama de Estrutura Composta", disponível em <http://www.fes.br/disciplinas/tpd/apa3/13Diagrama%20de%20Estrutura%20Composta.pdf>
Acedido a 02/01/2009.

Anjard, P. (1998), "Total Quality Management key concepts", *Work Study*, 7, 238–247.

Angeli (2008), disponível em <http://fit.faccat.br/~julianoangeli/linguagens/trabalhosql.html>
Acedido a 31/12/2008.

Aguiar, A., Faria, J., Flores, N., Mendes, P. "UML 2.0 – uma visão geral", disponível em http://paginas.fe.up.pt/~apaiva/teach/0809/ESOF_files/Visao.ppt Acedido a 02/01/2009.

Apcer (2003), Guia Interpretativo 9001:2000, disponível em http://www.apcer.pt/arq/fich/Guia_final.pdf Acedido a 27/12/2008.

Batini, C., Ceri, S., Navathe, S., (1992), "*Conceptual Database Design – An entity-relationship Approach*", The Benjamim/Cummings Publishing Company, Inc.

Beynon-Davies (2004), "Database Systems- Third Edition", Palgrave MacMillan.

Bouer (2005), "Ferramentas para o Controle e Melhoria de Qualidade", disponível em http://www.prd.usp.br/disciplinas/docs/pro2712-2006-Alberto_Gregorio/FERRAMENTAS%20OPERACIONAIS%20PARA%20A%20QUALIDADE.pdf
Acedido em 12/03/2009.

Booch, G., Rumbaugh, J., Jacobson, I. (1999), "The Unified Modelling Language User Guide", Addison-Wesley.

Bosch (2008), “Apresentação Oficial Bosch”, disponível em *Bosch Intranet*, Acedido a 25/02/2009.

Bosch (2008), “The 8D Problem – Solving Process”, disponível em *Bosch Intranet*, Acedido a 25/02/2009.

Bosch (2009), “Manual de Ajuda de Sistema de Gestão de Reclamações”, disponível em *Bosch Intranet*, Acedido a 25/02/2009.

Bosch (2009), “IQUIS e SupplyOn”, disponível em *Bosch Intranet*, Acedido a 25/02/2009.

Campagnaro, C., Rebelato, M., Rodrigues, A., Rodrigues, I., (2008), “Um estudo sobre métodos de análise e solução de problemas (MASP) na cadeia de fornecimento das montadoras automotivas nacionais”, egenep.

Carriço, J. (1996), *Desenho de Bases de Dados – Desenhar, Criar e Gerir Bases de Dados em Access*, Centro de Tecnologias de Informação.

Chen, H. (2001), “Benchmarking and Quality Improvement – A quality benchmarking deployment approach”; *International Journal of Quality & Reliability Management*, 6, 757-773.

Codd (1970), “A Relational Model of Data for Large Shared Data Banks”, IBM San Jose Research Laboratory.

Codd (1979), “Extending de Database Relational Model to Capture More Meaning”, IBM San Jose Research Laboratory.

Codd (1981), “Relational Database: A practical foundation for productivity”, IBM San Jose Research Laboratory.

Costa, M., Jimenez J. (2008), “Are companies that implement TQM better learning organizations? An empirical study.”, *Total Quality Management & Business Excellence*, 19, 1101-1115.

- Costa, j. (2004), “Transposição para o Modelo Relacional” disponível em www.geocities.com/engenheiroindustrial/doc/06Transpor.pdf Acedido a 30/03/2009.
- Constantine, L., Lockwood, L. (1999), “Software for use: A practical guide to the Models and Methods of Usage-Centered Design”, Addison-Wesley Professional.
- Cunha, P. João (2001), “Modelização Visual e UML Básico”, disponível em http://www2.egi.ua.pt/cursos_2003/files/APS/APS3-UML%20B%C3%A1sico.pdf Acedido a 02/01/2009.
- Dale, G.B. (1996), “Benchmarking on total quality management adoption: a positioning model”, *Benchmarking for Quality Management & Technology*, 1, 28-37.
- Davidson (2002), “Information Systems Development Techniques and Their Application to the Hydrologic Database Derivation Application”, disponível em <http://cadswes.colorado.edu/PDF/RiverWare/DavidsonLV2002.pdf> Acedido a 02/01/2009.
- Developer (2009), “Object Diagrams in UML”, disponível em <http://www.developer.com/design/article.php/2223551> Acedido a 02/01/2009.
- Dietrich, S., Urban D., “An advanced course in database systems – beyond relational databases”, Pearson Education.
- EdrawSoft (2009), disponível em http://www.edrawsoft.com/images/software/UML-Package_full.png, Acedido a 02/01/2009.
- Engin (2009), “The Visual Basic Programming Language”, disponível em <http://www.engin.umd.umich.edu/CIS/course.des/cis400/vbasic/vbasic.html> Acedido a 02/01/2009.
- Faria, J. (2002), “UML – Diagramas de Classes”, disponível em <http://twiki.fe.up.pt/pub/ASI1LCI0506/Asi1Documentos0506/UML-classes-redux.pdf>, Acedido a 02/01/2009.

- Faria, J. (2001), "UML – Diagramas de Objectos", disponível em <http://paginas.fe.up.pt/~jpf/teach/ES/UML/objectos.ppt> Acedido a 02/01/2009.
- Faria, J. (2001), "UML – Diagramas de Componentes e Modelização da Arquitectura Física", disponível em www.fe.up.pt/~jpf/teach/ES/UML/componentes.ppt Acedido a 02/01/2009.
- Faria J. (2001), "UML – Diagramas de Pacotes (Packages) e Modelização da Arquitectura Lógica", disponível em www.fe.up.pt/~jpf/teach/ES/UML/pacotes.ppt Acedido a 02/01/2009.
- Freytag V.P., Hollensen, S. (2001), "The process of benchmarking, benchlearning and benchaction", *The TQM Magazine*, 1, 25-33.
- Garden (2007), "8D Problem Solving" disponível em <http://www.stunell.com/8d-q8d.htm> Acedido a 12/03/2009.
- Harmon, P. (2008), "Managing to Learn: Using A3 management process to solve problems, gain agreement, mentor, and lead – Book Review" disponível em <http://www.bptrends.com/publicationfiles/03-03-09-BR-Mang%20to%20Learn-Shook-Harmon.pdf> Acedido a 12/03/2009.
- Hemsworth, D., Rodriguez, S., Bigood B. (2005), "Determining the impact of quality management practices and purchasing-related information systems on purchasing performance: A structural model", *The Journal of Enterprise Information Management*, 2, 169-194.
- Hernandez, M. (1997), "DataBase Desing for Mere Mortals", Assison-Wesley Developers Press.
- Juran, Joseph M., Godfrey, Blanton A. (1998), "Juran Quality Handbook", McGraw- Hill (5ª edição).
- Jacobson, I., Booch, G., Rumbaugh, J. (1999), "The Unified Software Development Process", Addison-Wesley.

- Lagunas, R., Hellman, B. (2007), "Organisational Advantages through the development of Information Systems", *Växjö University*.
- Larman, C. (2005), "Applying UML and Patterns – An introduction to object oriented Analysis and Design and Iterative Development", 3, Pearson Education.
- Laudon, K., Laudon, J. (2002), "Management Information Systems", 7, Prentice – Hall.
- Lucas et al. (2008), "Sistemas de Gestão de Bases de Dados", disponível em <https://aquila1.iseg.utl.pt/aquila/getFile.do?fileId=8207&method=getFile> Acedido a 03/03/2009.
- Karahasanovic, E., Lonn, H. (2007), "Selecting the best strategy to improve quality. Keeping in view the cost and other aspects", *Växjö University*.
- Madu, N.C., Kuei, C. H., Aheto, J., Winokur, D. (1994), "Integrating Total Quality Management in the Adoption of New Technologies", *Benchmarking for Quality Management & Technology*, 3, 52-66.
- Malvius (2007), "Information Management for Complex Product Development", *Royal Institute of Technology Sweden*.
- Martin, J. , Odell J. (1998), "Object- Oriented Methods: A Foundation", Prentice Hall PTR.
- Naiburg, J.E, Maksimchuk (2001), "UML for Database Design – Foreword by Grady Booch", Addison-Wesley.
- Newellrubbermaid (2009), "8D template", disponível em www.newellrubbermaid.com/newellco/downloads/CORP5QA-001.doc Acedido a 30/03/2009.
- ISO 9001:2008, "Quality Management Systems – Requirements", European Committee for standardization.

ISO 9000:2000,"Conceitos e Terminologia", European Committee for standardization.

Oakland, Jonh S, Sohal, Amrik S.(2001), "Total quality management : text with cases", Pacific Rim Edition.

Omachonu Vincent K., Ross, Joel E. (2004), "Principles of Total Quality Management third edition", CRC Press.

Rahman Z. ,Siddiqui J., 2006, "Exploring total quality management for information systems in Indian firms Application and benefits".

Redman, T. (1996), "Data quality for the information Age", Artech House.

Rodrigues, S., Hemsworth, D. (2005), "A Structural Analysis of the Impact of Quality Management Practices in Purchasing on Purchasing and Business Performance", *Total Quality Management*, 2, 215–230.

Salvada, P. (2008), "Metodologia A3" disponível em http://www.leanthinkingcommunity.org/livros_recursos/PDCA%20Metodologia%20A3%20segundo%20Pedro%20Salvada.pdf Acedido a 12/03/2009.

Shneiderman, B., Plaisant, C. (2005), "Designing the User Interface",4, Pearson Education.

Stanley, L.L., Wisner b, J.D. (2001), "Service quality along the supply chain: implications for purchasing" , *Journal of Operations Management*,19, 287–306.

Stahl, F. (2006), "Instantiation of an ISO 9000 compliant quality assurance process", *Växjö University*.

Sinfic (2008), "Solução Modelização em UML" , disponível em <http://www.sinfic.pt/SinficWeb/docs/22446.pdf> Acedido a 02/01/2009.

Sinfic (2005), "Sistemas de Gestão da Qualidade (ISO 9001:2000)", disponível em <http://www.sinfic.pt/PortalWeb/newsletter/sinfic/Newsletter44/Dossier2.html> Acedido a 02/01/2009.

Sobek, D., Jimmerson, C., (2004) "A3 Report: Tool for Process Improvement", disponível em

<http://www.lean.org/Community/Registered/ArticleDocuments/A3%20report%20with%20map%20Tools%20for%20process%20improvement.pdf> Acedido a 12/03/2009.

Sousa, Luís (2008), "Concepção de Uma BD para Apoio ao Desenvolvimento de Novos Produtos", *Universidade de Aveiro*.

Warren (2002), "8D process" disponível em http://www.isixsigma.com/dictionary/8_D_Process-170.htm Acedido a 12/03/2009.

Tim (2003), "UML 2.0 – Adoptado Oficialmente na região de Paris", disponível em http://paginas.ispgaya.pt/~msantos/es_artigos_tecnicos_1/21_UMA20_Adoptado_Oficialmente_na_reuniao_de_Paris.pdf Acedido a 02/01/2009.

Torres, M. (2005), "Apontamentos de Programação em Visual Basic", *Universidade Fernando Pessoa*.

Turtle (2006), "A UML profile for the modeling and Formal Verification for Time Constrained Systems", disponível em http://labsoc.comelec.enst.fr/turtle/web_iod01.gif Acedido a 02/01/2009.

Visual Paradigm (2009), "UML 2 Diagrams", disponível em <http://www.visual-paradigm.com/VPGallery/diagrams/index.html>. Acedido a 02/01/2009.

Villiers (2008), "The illustrated Lean CookBook", disponível em <http://www.scribd.com/doc/5396036/The-Illustrated-Lean-Agile-and-World-Class-Manufacturing-Cookbook-2008-08-20> Acedido a 12/03/2009.

UML Elements (2009), "UML Basics", disponível em http://docs.kde.org/stable/en_GB/kdesdk/umbrello/uml-elements.html, Acedido a 02/01/2009

UML org (2009), "Class Diagram", disponível em <http://www.uml.org.cn/object/images/classDiagramInitial.jpg> Acedido a 02/01/2009.

Yang, L., Vernussi, A., Feng, L., Esparham, S. (1998), "Methodologies for Development" disponível em <http://www.otal.umd.edu/guse/methodologies.html> Acedido a 08/03/2009.

Zairi M. (1994), "Benchmarking: The Best Tool for Measuring Competitiveness", *Benchmarking for Quality Management & Technology*, 1, 11-24.

6. Anexos

Anexo 1.Exemplo de Template 8D (Adaptado de newellrubbermaid (2009)).

				CA #:
DMR # :	Part #:	Description:	Initiated by:	
			Date Opened	
D0 Symptom(s)				
D0 Emergency Response Action(s)			% Effective	Date Implemented
Verification/Validation completed by/date:				
D1 Team (Name, Dept., Phone) Champion: Team Leader: Team Members)			D2 Problem Statement	
D3 Interim Containment Action(s)			% Effective	Date Implemented
Verification/Validation completed by/date:				
D4 Root Cause(s)				% Contribution
Verification by:				
D5 Chosen Permanent Corrective Action(s)				% Effective
Verification/Validation completed by/date:				
D6 Implemented Permanent Corrective Action(s)				Date Implemented
Verification/Validation completed by/date:				% Effective
D7 Prevent Actions			Date Implemented	
Verification/Validation completed by/date:				
D7 Systemic Prevent Recommendations			Responsibility / Date	
D8 Team and Individual Recognition			Date Closed	Reported By

Anexo 2. Classificação e Descrição dos diferentes tipos de Diagramas UML (UML 2.0).

Diagramas de Comportamento (Diagramas de Atividades, de Estado e de Casos de utilização) – Demonstram os aspectos comportamentais e de reacção do sistema (Aguiar et al., 2007). Representam os aspectos dinâmicos do sistema, ou processo de negócio e suas características comportamentais (Tim, 2003).

Diagramas de Actividade

Estes diagramas representam processos de negócio (Tim, 2003, Aguiar et al.2007), comportamento de algoritmos e casos de utilização sendo inspirados nos fluxogramas (Aguiar et al. 2007). Os diagramas de actividade estão sempre associados a uma classe, operação ou caso de utilização, como se pode ver exemplificado em Sousa, (2008).

Através deles podem-se representar acções e actividades. Normalmente as transições ocorrem quando as acções, ou actividades, se concluem (Aguiar et al. 2007).

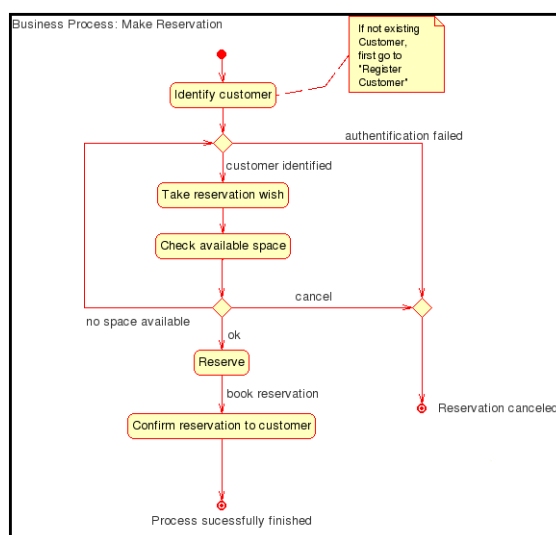


Figura 45. Diagrama de Atividades (UML Elements (2008)).

No exemplo (ver figura 45) pode-se verificar que os autores têm razão, existindo uma semelhança deste tipo de diagramas com os fluxogramas. Neste caso temos as actividades: Identify Customer, Take Reservation Wish, Check Available Space, Reserve, Confirm Reservation to Customer. O processo utilizado no exemplo, tal como os autores referem, traduz-se num processo de negócio de um restaurante. Naiburg & Maksimchuk (2001) afirmam que estes diagramas são essenciais em muitas questões, como por exemplo:

- ✓ Compreensão do negócio.
- ✓ Identificação de áreas de negócio que necessitem de alteração.
- ✓ Descoberta de redundâncias nos processos de negócio.
- ✓ Descoberta de gargalos nos processos de negócio.
- ✓ Identificação de actividades que possam ser melhor desempenhadas interna ou externamente.
- ✓ Estabelecimento de necessidades de informação de uma actividade ou caso de utilização particular.

Diagramas de Estado

Os Diagramas de Estado descrevem os estados em que podem estar um objecto, ou interacção, bem como as transições entre estados ao longo da sua existência (Aguiar et al. 2007, Tim, 2003).

Por exemplo, na seguinte imagem o objecto, durante a sua vida, pode estar num dos seguintes estados: *Ready*, *Listen*, *Working* ou *Stopped* (ver figura 46).

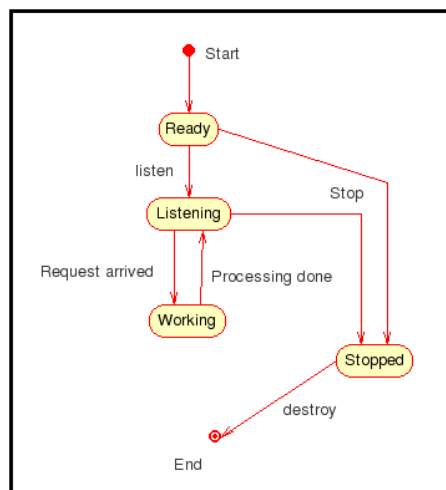


Figura 46. Diagrama de Estado (UML Elements (2008)).

O objecto muda de estado quando ocorrem certos eventos. Por exemplo, existe a passagem de *Listening* para *Working* quando acontece o evento *Request Arrived*.

Diagramas de Casos de utilização

Através dos Diagramas de casos de utilização podem-se descrever relações entre actores, casos de utilização de um sistema (Aguiar et al. 2007) e as suas interrelações (Tim, 2003).

Estes diagramas são fundamentais na definição das fronteiras dos sistemas sendo utilizados nas tarefas de especificação de requisitos e modelização de processos de negócio (Aguilar et al. (2007), Naiburg & Maksimchuck (2001)). Nestes diagramas os actores, são os utilizadores do sistema e os casos de utilização correspondem às funcionalidades disponíveis aos actores (ver figura 47).

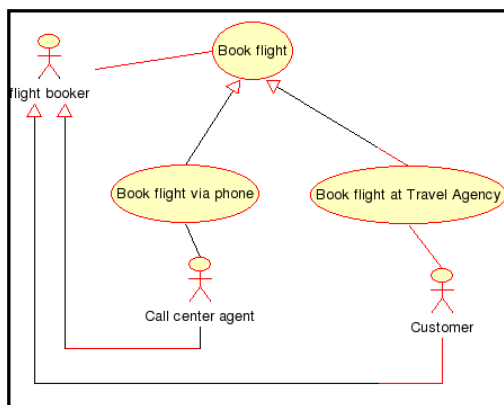


Figura 47. Diagrama de Casos de utilização (UML Elements (2008)).

No exemplo apresentado o actor *Customer* pode, por exemplo, executar a acção *Book Flight at Travel Agency*.

Diagramas de Interações de Objectos (Diagramas de Comunicação, de Interação, de Sequência e de Tempo) – Enfatizam a interação entre os objectos (Aguilar et.al., 2007, Tim, 2003), mostrando como interagem uns com os outros. Pertencem aos diagramas Comportamentais.

Diagramas de Comunicação

Este tipo de diagramas mostra as instâncias de classes, as suas interrelações e os fluxos de mensagem (Tim, 2003).

Dão ênfase à organização estrutural dos objectos que enviam e recebem mensagens (Tim, 2003), utilizando números sequenciais, com diferentes níveis hierárquicos (Aguilar et al., 2007). Neste caso as relações entre os objectos são o principal contributo (ver figura 48).

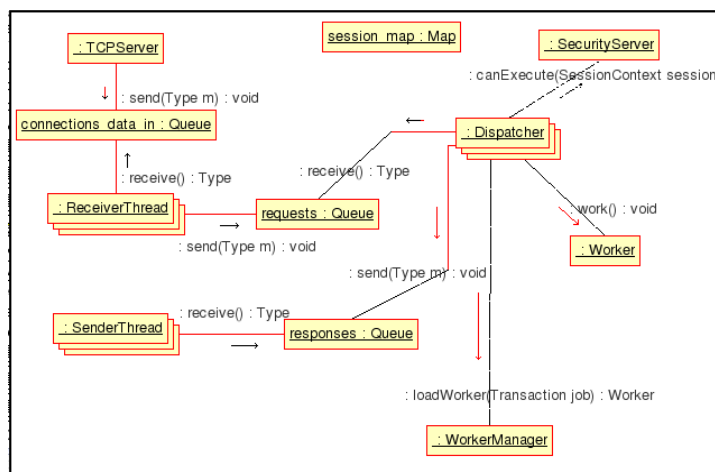


Figura 48. Diagrama de Comunicação (UML Elements (2008)).

Nestes diagramas, como se pode verificar pelo exemplo, podem-se apresentar facilmente fluxos e processos, com as mensagens que passam de um objecto para outro.

Diagramas de Interação

Os diagramas de interação são uma variante dos diagramas de actividade (Tim, 2003). São utilizados para quando se pretende descrever o comportamento de vários objectos em apenas um caso de utilização.

Através destes diagramas pode-se verificar a sequência de eventos que ocorre num caso de utilização. Um Caso de utilização, *Connection*, pode envolver os passos que são detalhados no exemplo seguinte (ver figura 49).

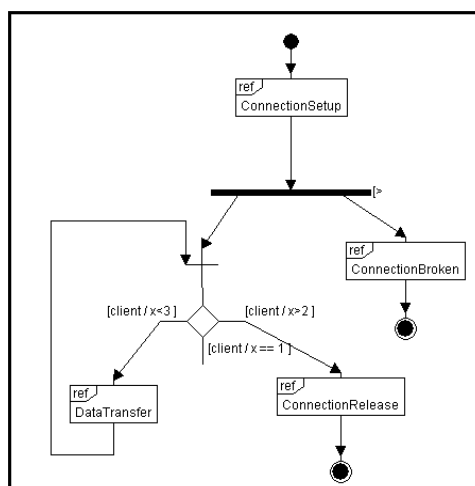


Figura 49. Diagrama de Interação (Turtle (2006)).

Diagramas de Sequência

Aguiar et al. (2007) dizem que estes diagramas ilustram interações entre objectos num determinado período de tempo. Mostram, então, uma interacção entre objectos com uma determinada ordem e tempo na troca de mensagens. Dá ênfase à ordenação no tempo das mensagens (Aalst & Hee, 2002).

Os objectos são representados por linhas a tracejado na vertical, estando o seu nome no topo da mesma. O tempo aumenta de cima para baixo. As mensagens têm o formato de setas. Pode-se ver um exemplo destes diagramas na Figura 50. No diagrama exemplificado verifica-se que a mensagem *work()* é a ultima a ser enviada por ser a que se encontra mais em baixo, e a *receive()* é a primeira.

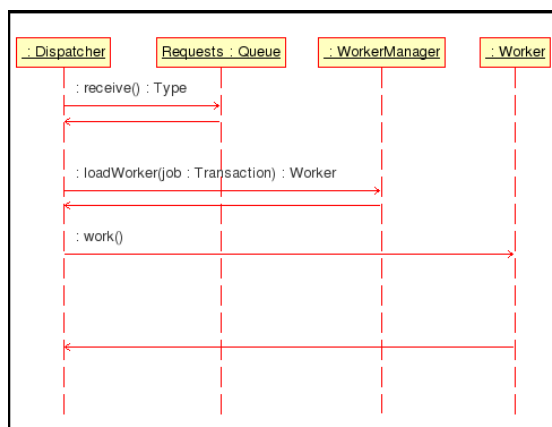


Figura 50. Diagrama de Sequência (UML Elements (2008)).

Diagramas de Tempo

Estes Diagramas são utilizados, normalmente, para mostrar a mudança de estado de um objecto ao longo do tempo em resposta a acontecimentos externos (Tim, 2003).

Sousa (2008) exemplifica que se pode observar o comportamento dos objectos num determinado período de tempo (ver figura 51).

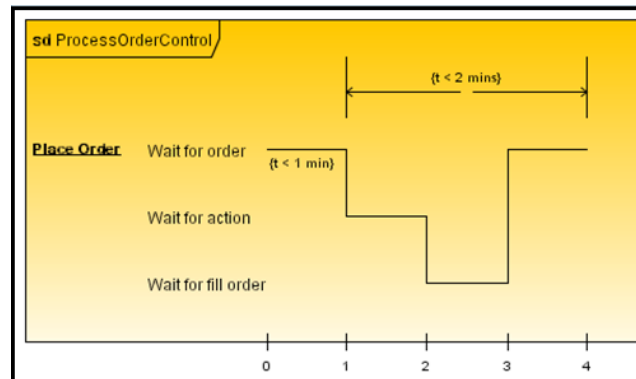


Figura 51. Diagrama de Tempo (Visual Paradigm (2009)).

Estes diagramas são bastante parecidos com os de sequência, como se pode verificar na figura 51. No entanto, aqui, o tempo aumenta da esquerda para a direita, e não de cima para baixo.

Diagramas de Estrutura (Diagramas de Classes, de Estruturas Compostas, de Componentes, de Objectos, de Pacotes e de Instalação) – Especificação estrutural dos elementos independentes do tempo, ou seja, dos aspectos estáticos do sistema (Aguiar et al. 2007, Tim, 2003).

Diagramas de Classes

Conjunto de elementos de modelo estáticos, nomeadamente classes e tipos, os seus conteúdos e as suas relações (Tim, 2003, Naiburg & Maksimchuk, 2001).

Aguiar et al.(2007) descrevem este tipo de diagramas como sendo a estrutura de um sistema reflectindo, em particular, as entidades, estruturas internas, e relações existentes.

Faria (2002) diz que estes diagramas servem para modelizar o vocabulário de um sistema, do ponto de vista do utilizador / problema ou do programador / solução.

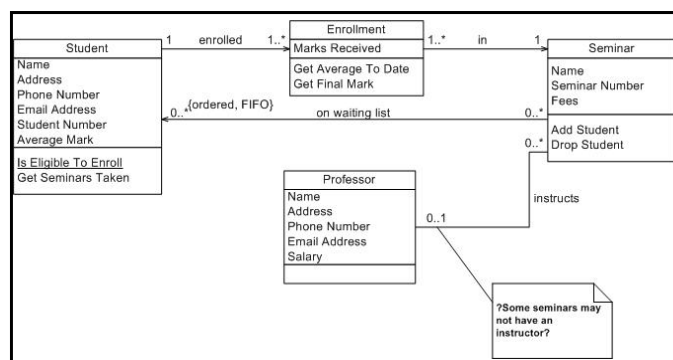


Figura 52. Diagrama de Classes (UML org (2009)).

Através do exemplo na figura 52 pode-se verificar a estrutura de um dado sistema e as relações existentes no mesmo.

Diagramas de Estruturas Compostas

Apa (2008) refere-se a estes como sendo Diagramas para modelizar colaborações. As colaborações representam um conjunto de elementos que interagem entre si no seu interior (Apa, 2008)

O mesmo autor diz, também, que estes diagramas podem descrever a colaboração interna de classes, ou ser usados para especificar funcionalidades.

Seguidamente encontra-se um exemplo deste tipo de Diagramas (ver figura 53).

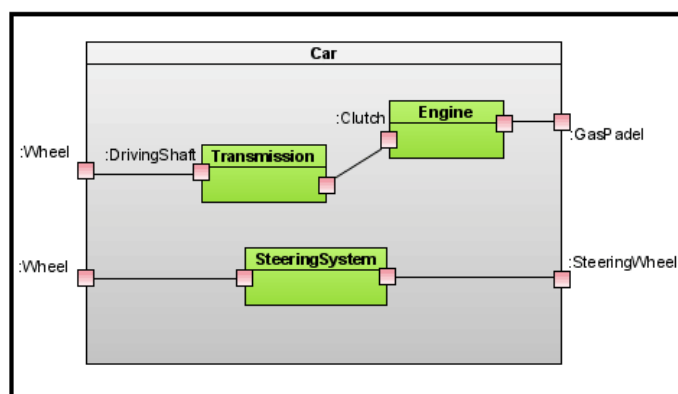


Figura 53. Diagrama de Estruturas Compostas (Visual Paradigm (2009)).

Diagramas de Componentes

Tim (2003) diz que estes diagramas apresentam os componentes que compõem uma aplicação, sistema, ou empresa.

Faria (2001) diz que servem para capturar a estrutura física da implementação e são construídos como parte da especificação da Arquitectura. O mesmo autor afirma que os principais objectivos destes diagramas são:

- Organizar o código fonte (ambiente de desenvolvimento).
- Construir uma versão executável (ambiente de produção).
- Especificar uma BD física.

Afirma também que contém componentes, interfaces e relações entre componentes. Aguiar et al. (2007) afirmam que estes servem para descrever as dependências entre componentes de software, incluindo componentes de código fonte, código binário e executáveis.

Seguidamente encontra-se exemplificado um Diagrama deste tipo (ver figura 54).

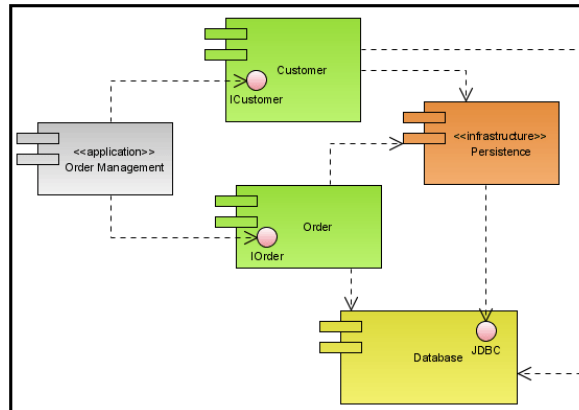


Figura 54. Diagramas de Componentes (Visual Paradigm (2009)).

Diagramas de Objectos

Mostram os objectos e as suas relações num determinado momento, sendo normalmente um caso especial de um diagrama de classes ou de comunicação (Tim, 2003). Retratar cenários de possíveis concretizações de um sistema (Aguiar et al., 2007).

Estes diagramas são semelhantes ao de classes mas retratam um momento de execução do sistema.

No exemplo a seguir, pode-se verificar um Diagrama deste tipo. Neste caso verifica-se que univName = "University of Chicago" e nameOfStudent = "Sam". Verifica-se portanto que o diagrama reflecte o funcionamento num determinado instante (ver figura 55).

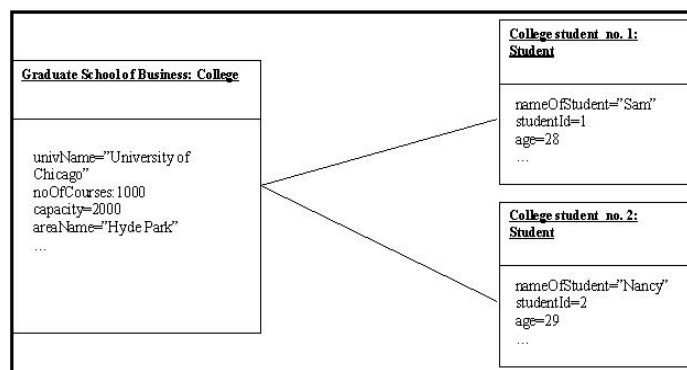


Figura 55. Diagrama de Objectos (Developer (2009)).

Diagramas de Pacotes

Tim (2003) afirma que estes diagramas mostram como os elementos do modelo estão organizados em pacotes, bem como as dependências entre pacotes.

Faria (2001) refere que um diagrama de pacotes mostra pacotes e relações entre pacotes. Este mesmo autor afirma que não existem propriamente diagramas de pacotes em UML e que pacotes, e relações entre os mesmos, aparecem noutros diagramas, de acordo com cada tipo de pacote:

- ✓ Pacotes de classes – em diagramas de classes.
- ✓ Pacotes de componentes – em diagramas de componentes.
- ✓ Pacotes de nós – em diagramas de distribuição.

Pacotes de casos de utilização – em diagramas de casos de utilização.

O autor refere que um pacote é um mecanismo de agrupamento genérico.

Seguidamente encontra-se um exemplo deste tipo de Diagrama (ver figura 56).

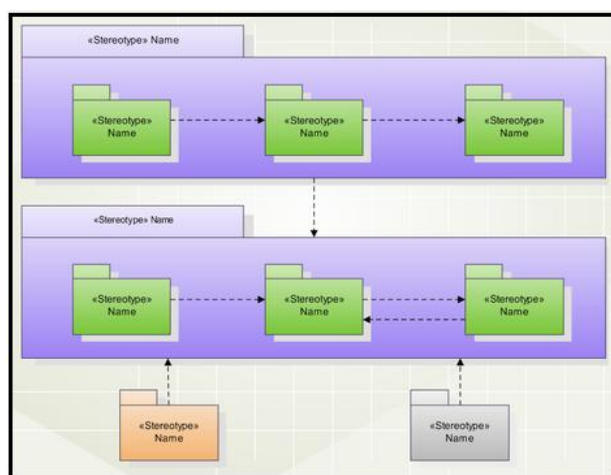


Figura 56. Diagrama de Pacotes (EdrawSoft (2009)).

Diagramas de Instalação

De acordo com Tim (2003) estes diagramas mostram a arquitectura de execução dos sistemas que inclui nós, ambientes de execução, hardware e software.

Aguiar et al. (2007) dizem que descrevem a configuração de elementos de suporte ao processamento, software e objectos.

Basicamente estes diagramas descrevem o software e hardware do sistema e a sua interacção com os restantes elementos do memo, retratando assim a sua arquitectura.

De acordo com Visual Paradigm (2009) estes diagramas auxiliam a modelizar os aspectos físicos do sistema.

Seguidamente apresenta-se um exemplo deste tipo de diagrama (ver figura 57).

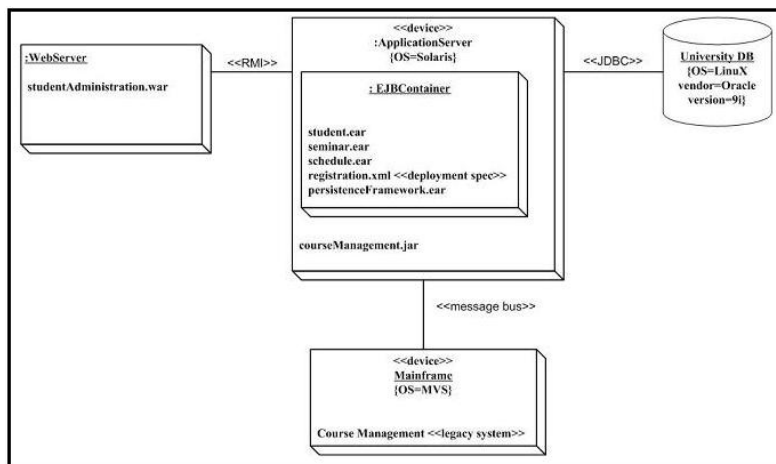


Figura 57. Diagrama de Instalação (Aguiar et al (2007)).

Anexo 3. Ferramenta A3 do Projecto.

Anexo 4. Dicionário de dados – diagrama de classes.

Classe	Descrição da Classe	Atributo	Descrição do Atributo	Domínio (SGBD Access)
8D	Classe 8D associada à classe Complaints e 8D_supplier_actions	8d_code	Código identificativo do 8D	integer
		request_date	Data de pedido do 8D	date
		efficiency	Verificação da eficiência	boolean
8D_supplier_action	Classe 8D_supplier_actions associada à classe 8D e 8D_action_types	8d_sa_code	Código identificativo da acção	integer
		8d_action_description	Descrição da acção tomada	string
		action_validation	Validação da acção	boolean
		8d_action_description_date	Data da acção	date
8D_action_type	Classe 8D_action_types associada à classe 8D_supplier_actions	8d_action_type	Tipo de acção do 8D	integer
		8d_action_description	Descrição do tipo de acção	string
Complaint	Classe Complaints, associada à classe 8D, SupplierActions, Attachments, Origins, Material, Status e ComplaintImportance.	complaint_number	Código identificativo da reclamação	integer
		notification	Notificação associada	integer
		FD	Data de fabrico do componente	integer
		complaint_date	Data de pedido da reclamação	boolean
		problem_description	Descrição do problema	string
		among_of_defect_units	Quantidade de defeitos	integer
		Origin_result	Resultado avaliado pela Origem	boolean
		notes	Observações sobre o problema	date
		complaint_title	Título da reclamação	string
		result	Resultado final da reclamação	boolean
Complaint_importance	Classe Complaints_importance associada à classe Complaints	complaint_importance	Importância da reclamação	string
		description	Descrição da importância	string
Action_type	Classe do tipo Action_Types associada à classe	action_type	Tipo de acção	string

	SupplierActions.	action_description	Descrição da acção	string
Supplier_action	Classe Supplier_Actions associada à classe Action_Types e Complaints.	action	Código identificativo da acção	integer
		action_detail	Detalhe da acção	string
Attachment	Classe Attachments associada à classe Complaints	attachments_code	Código identificativo do anexo	integer
		attachment	Anexo	image
		attachment_description	Descrição do anexo	string
Image	Classe Images associada à classe Complaints	image_code	Código identificativo da imagem	integer
		image	Imagem	image
		description	Descrição	string
Origin	Classe Origins associada à classe Complaints	origin_code	Código identificativo da origem	string
		origin_name	Nome da origem	string
Material	Classe Material associada às classes Complaints, Supplier e Failure_Codes	material	Código identificativo do material	integer
		material_description	Descrição do material	string
Status	Classe Status associada à classe Complaints	complaint_status	Status das reclamações	string
		status_description	Descrição do status	string
Supplier	Classe Suppliers associada à classe Material, Responsibles, Class e Material_groups	supplier	Código identificativo do fornecedor	integer
		supplier_name	Nome do fornecedor	string
		supplier_email	Email do fornecedor.	string
Classe	Classe Classes associada à classe Suppliers	class	Classe do material	string
		class_description	Descrição da classe	string
Responsible	Classe Responsibles associada à classe Suppliers e Puq_replacement	puq_code	Código identificativo do responsável PUQ	string
		name	Nome do responsável PUQ	string
		email	Email do responsável PUQ	string
Puq_replacement	Classe Puq_replacement associada à classe	replacement	Código identificativo do substituto PUQ	string

	Reponsibles	replacement_name	Nome do substituto	string
Failure_code	Classe Failure_codes associada à classe Material e Failure_classifications	failure_code	Código de falha	integer
		failure_description	Descrição da falha	string
Failure_classification	Classe Failure_classifications associada à classe Failure_codes	component_gr	Códigos do grupo de componentes	string
		component_gr_description	Descrição dos códigos do grupo de comp.	string

Anexo 5.Diagrama de Classes – em versão A3.

Anexo 6. Modelo Relacional – em versão A3.

Anexo 7. Visual Basic.

O Visual Basic é uma linguagem de Programação para desenvolvimento de aplicações no Windows. Através desta ferramenta é possível desenvolver aplicações complexas em relativamente pouco tempo.

Com esta ferramenta primeiro define-se a componente da interface e, após isso, define-se o código necessário.

Esta linguagem é baseada em eventos (acções) do utilizador. Normalmente quando o utilizador clica num botão, move o rato, ou desenvolve outro tipo de acção os procedimentos são chamados e o código é executado.

Torres (2005) afirma, precisamente, que esta linguagem implementa o modelo orientado aos eventos respondendo muitas das funções a eventos do utilizador.

O mesmo autor afirma ainda que esta linguagem possui características específicas para desenvolver rápida e facilmente a parte gráfica da interface de um programa (GUI – *Graphical User Interface*).

Uma outra grande vantagem desta programação é que muito do código é gerado automaticamente à medida que se constrói ambiente gráfico, o que poupa muito tempo ao programador. Noutras línguas tudo tem que ser programado em código.

Visual Basic também pode ser utilizado em aplicações de BD's, servindo de interface para recolha de dados e apresentação de informação sendo compatível, também, com o uso de SQL. De acordo com Engin (2009) esta ferramenta de programação apresenta uma metodologia baseada em 3 passos, na sua utilização: desenho da aplicação – interface, estabelecimento das propriedades dos objectos e Escrita do código para tarefas específicas.

Esta abordagem é coincidente com o que já foi dito, uma vez que pressupõe a construção da aplicação antes de se existir inserção de código e associa o código à execução de tarefas específicas.

Verifica-se portanto que esta ferramenta é útil, uma vez que a criação aplicações se dá de forma rápida, e permite programar apenas os eventos necessários.

Através da figura 58 exemplifica-se o código associado ao clique de um botão. Clicando no respectivo botão o programa corre e executa as condições codificadas.

```
Private Sub Comando16_Click()  
On Error GoTo Err_Comando16_Click  
    DoCmd.GoToRecord , , acNext  
Exit_Comando16_Click:  
    Exit Sub  
Err_Comando16_Click:  
    MsgBox Err.Description  
    Resume Exit_Comando16_Click  
End Sub
```

Figura 58. Exemplo de código Visual Basic.

Anexo 8.Exemplo de relatório gerado através de consulta.

8D - Pending By Supplier FORNECEDOR 1

complaint number

Problem Description: fora de especificação. 1

request date	<i>dated1</i>	<i>dated2</i>	<i>dated3</i>	<i>dated4</i>	<i>dated5</i>	<i>dated6</i>	<i>dated7</i>	<i>dated8</i>
27-06-2008	27-06-2008	27-06-2008	27-06-2008	27-06-2008	27-06-2008	13-07-2008	28-10-2008	

Problem Description: fora de especificação. 2

request date	<i>dated1</i>	<i>dated2</i>	<i>dated3</i>	<i>dated4</i>	<i>dated5</i>	<i>dated6</i>	<i>dated7</i>	<i>dated8</i>
08-10-2008								

Problem Description: fora de especificação. 3

request date	<i>dated1</i>	<i>dated2</i>	<i>dated3</i>	<i>dated4</i>	<i>dated5</i>	<i>dated6</i>	<i>dated7</i>	<i>dated8</i>
21-04-2008	23-04-2008	23-04-2008	23-04-2008	23-04-2008	23-04-2008	30-04-2008		

Problem Description: fora de especificação. 4

request date	<i>dated1</i>	<i>dated2</i>	<i>dated3</i>	<i>dated4</i>	<i>dated5</i>	<i>dated6</i>	<i>dated7</i>	<i>dated8</i>
29-07-2008	31-07-2008	31-07-2008	31-07-2008	24-10-2008	24-10-2008	24-10-2008		

Anexo 9. Exemplo de Processo do Manual.

“6.3- Processo de Inserção de Fotos e Anexos:

1. Clicar com na tecla direita do rato sob a área em que a foto será inserida como consta na imagem seguinte (ver imagem 26).

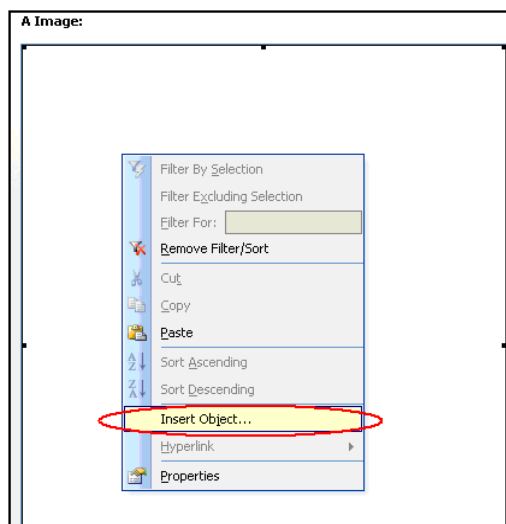


Imagem 26. Menu de Inserção de Imagens no Sistema.

2. Na barra de opções que surge clicar em **Insert Object**. Logo após surge o seguinte menu (ver imagem 27)

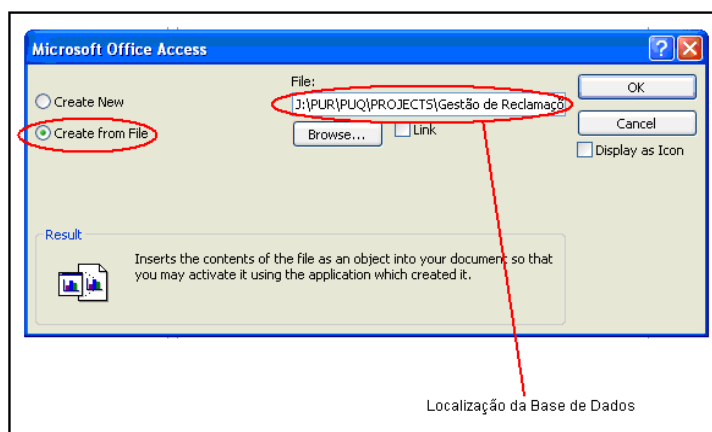


Imagem 27. Localização da Imagem para Inserir no Sistema.

3. Escolher a opção **Create from File**, e depois inserir o directório da imagem na caixa de texto **File**. Clicar ok, finalizando-se o processo.”

Fonte: Bosch (2009).

Anexo 10. SQL (*Structured Query Language*).

• Linguagem SQL

Embora existam SGBD que possibilitam a construção das BD's através de Interfaces gráficas, como por exemplo o MS-Access, outras só são possíveis de desenvolver através da utilização de uma linguagem específica a que se dá o nome de SQL (*Structured Query Language*).

A linguagem SQL é um meio através do qual se pode efectuar a implementação do modelo Relacional. Angeli (2008) diz que a linguagem SQL tanto é de definição como de manipulação de dados, podendo-se definir e construir relações (tabelas), como manipulá-las de forma a obter resultados pretendidos (consultas). O mesmo autor diz que a SQL implementa os conceitos definidos no Modelo Relacional, podendo-se através dela:

- ✓ Criar, Alterar e Remover componentes de uma BD (por ex: tabelas)
- ✓ Inserir, Alterar e Apagar dados.
- ✓ Efectuar consultas.
- ✓ Controlar o acesso dos utilizadores à BD.
- ✓ Garantir a integridade dos dados.

Dentro da SQL distingue-se a DML (*Data Manipulation Language*) e a DDL (*Data Definition Language*) (Dietrich & Urban, 2005). De seguida descreve-se sucintamente as vertentes da SQL segundo estes autores:

A DDL inclui argumentos que modificam e eliminam definições das tabelas, sendo responsáveis por actualizar a BD. A instrução *Alter Table* pode ser utilizada para esse efeito, recorrendo-se para isso às instruções *Add* e/ou *Drop*., como no exemplo seguinte:

Alter table <table_name> <alter_table_action>

<alter_table_action> ::=

Add [column] <column_definition>

Drop [column] <column_name>

Existem também instruções para efectuar restrições de integridade como, por exemplo, <column_constraint> ou <table_constraint>.

Por outro lado podem-se criar tabelas através da DDL utilizando para isso a instrução *Create*, como se encontra no exemplo seguinte:

```
Create table <table_name>
( <var1> <var_type> primary key,
  <var2> <var_type> no null, ...)
```

A DML, por outro lado, inclui argumentos que inserem, eliminam e actualizam dados numa BD, como se pode verificar, respectivamente, nos seguintes exemplos:

```
Insert into <table_name>
Value (<value>)
```

```
Delete from <table_name> [where <search_condition>]
```

```
Update <table_name> set <set_clause>
[where <search_condition>]
Set ::= <column_name> = <row_value_designator>
```

E por outro lado possui argumentos que permitem questionar a BD, como permitindo assim efectuar consultas, como se verifica no exemplo seguinte:

```
Select <value>
From <table_name>
Where <search_condition>
Order by <value>
```

Apesar de alguns SGBD possuírem modos de construção que dispensam conhecimento de SQL, ao implementar desta forma pode-se sempre transitar facilmente para um outro SGBD bastando para isso implementar o modelo definido em SQL.

Estes anexos só estão disponíveis para consulta através do CD-ROM.
Queira por favor dirigir-se ao balcão de atendimento da Biblioteca.

Serviços de Biblioteca, Informação Documental e Museologia
Universidade de Aveiro